

ICESat-2 Algorithm Theoretical Basis Document for the Atmosphere, Part I: Level 2 and 3 Data Products

Steve Palm

Yuekui Yang

Ute Herzfeld

David Hancock

Version 8.5

September 9, 2021

Contents

Release Notes	4
1 Introduction and Background	10
2 Corrections to the Raw Atmospheric Profiles.....	16
2.1 Dead Time Correction	16
2.2 Correction for Raw Profile Shifting	21
2.3 Transmit Echo Pulse.....	21
3 L2A Product: Normalized Relative Backscatter (ATL04)	23
3.1 L2A Required Inputs.....	23
3.2 L2A Outputs	24
3.3 NRB Computation	30
3.3.1 Molecular Backscatter Computation	30
3.3.1.1 Molecular Transmission	34
3.3.2 Molecular Scattering Folding Correction	34
3.3.2.1 Error Analysis of Molecular Contribution	37
3.3.3 Ozone Transmission Computation.....	37
3.3.4 Background Computation	38
3.3.4.1 Background Method 1.....	38
3.3.4.2 Background Method 2.....	41
3.3.4.3 Background Method 3.....	41
3.3.4.4 Theoretical Background	44
3.3.5 Surface Signal	45
3.3.6 Vertical Height Adjustment.....	48
3.3.7 Calculation of the Calibration Coefficient.....	49
3.3.7.1 Theory	49
3.3.7.2 Calibration Algorithm using the Atmosphere	50
3.3.7.3 Calibration Algorithm using Surface Reflectance	58
3.3.7.4 Calibration Error and Confidence	61
3.4 Cloud Folding Flag	61
3.4.1 Cloud Folding Flag Part II	62
4 Level 3 Product (ATL09)	63
4.1 L3 Required Inputs	63

4.2 L3 Outputs	64
4.3 Calibrated, Attenuated Backscatter Profiles	72
4.4 Layer Heights and Flags from Backscatter Profiles	73
4.4.1 Layer Integrated Attenuated Backscatter	74
4.5 Cloud/Aerosol Discrimination	74
4.5.1 Background	74
4.5.2 Layer Discrimination Confidence Factor Algorithm	75
4.5.3 Implementation of the Layer Discrimination Algorithm	76
4.6 Blowing Snow	83
4.6.1 Blowing Snow Layer Height	83
4.6.2 Blowing Snow Layer Optical Depth	86
4.6.3 Blowing Snow Layer Confidence	87
4.6.4 Blowing Snow Probability	88
4.7 Apparent Surface Reflectance (ASR)	89
4.7.1 Cloud Detection using ASR	91
4.7.2 ASR Cloud Detection Algorithm Implementation	95
4.7.2.1 ICESat-2 Clear Sky ASR Climatology	96
4.7.2.2 Cloud Detection Using Apparent Surface Reflectance	98
4.8 Ocean (or Open Water) Surface Reflectivity	101
4.9 Total Column Optical Depth Using ASR	102
The Column_OD_ASR_QF will be computed using the NOAA flag and the IGBP flag.	104
5.0 Consolidated Cloud Flag	104
6.0 Quality Assessment	105
6.1 Calibration	105
6.2 Background	105
6.3 Surface Signal	105
6.4 Calibrated Attenuated Backscatter	105
6.5 Layer Heights	105
6.6 Apparent Surface Reflectance	105
6.7 Ocean Surface Reflectance	106
6.8 Total Column Optical Depth	106
6.9 Multiple Scattering Warning Flag	106

7 Product Quality Parameters.....	106
Appendix A.....	107
Appendix B.....	109
Appendix C.....	110
Appendix D.....	113
References	116

Release Notes

This version, **8.5, dated September 9, 2021** corresponds to ASAS 5.5 (product release 005). All changes in **red**. The major change is the revision to the true surface reflectance map being used for ASR cloud detection and total column optical depth (over land, ice and sea ice). Prior versions used a monthly climatology based on GOME surface reflectance observations. This has now been replaced by a surface reflectance map constructed from ICESat-2 measurements of Apparent Surface Reflectance (ASR) using profiles devoid of cloud or aerosol. This is described in section 4.7.2. Additionally, the POD/PPD flag was added to the ATL04 and ATL09 products. This is described in section 7 and Tables 3.1 and 4.1. It was determined that the deadtime correction algorithm used in prior versions had a problem most notably over Antarctica in winter. A second algorithm, described by equation 2.1 is now being used. This change did not require a code change, but rather a change to a control variable (dtime_select=2).

Prior Version Updates:

This version, **8.4, dated January 15, 2021** corresponds to ASAS 5.4 (product release 004). All changes/additions are in **red**. The major change is **1)** a new surface detection algorithm which now reports surface signal width in bins and the confidence of the surface detection (Section 3.3.5), and **2)** a new cloud/aerosol discrimination algorithm (Section 4.5). Additional changes include **3)** Transmit Echo Pulse (TEP) removal that now uses a Poisson distributed value based on the average of the bins above and below the TEP bin to replace the TEP bin (Section 2.3). **4)** Changes to various constants that control the magnitude of the molecular folding correction (Section 3.3.2), **5)** improved calibration in the South Atlantic Anomaly region and changes to calibration related parameters and values of *calib_solar_elev_min* and *calib_solar_elev_max* (Section 3.3.7.2 and Table 3.6), **6)** Modified the blowing snow detection algorithm, added the parameter *max_bsnow_cab* and changed the value of *bsnow_top_scale*, **7)** Modified the value of F in equation 4.7 (Apparent Surface Reflectance), **8)** Changed the value of phi land and phi ocean (Section 4.7.2.3)

For this version, 8.3, **dated February 14, 2020** corresponds to ASAS 5.3 software which is used to produce the version 003 atmospheric products. This version of the ATBD is really just an update to the 8.2 version from August, 2019. In this and that ATBD, we have modified section 3.3.2 – Molecular Scattering Folding Correction. Here, Table 3.2a was added to identify various values of the parameter “alpha” that is used in equation 3.15. Changed the values of ancillary constants *grnd_thresh1_sf* and *grnd_thresh2_sf* in Table 3.3.5. Changed the value of the ancillary constants *cal_solar_elev_min* and *cal_solar_elev_max* in section 3.3.7.2 (page 53). Changed the description of the application of the calibration constant to the NRB profiles in section 4.3 to not involve any prior or following granules. Added *bsnow_intensity* to the ATL09 product – see section 4.5.1. Added discussion of *bsnow_sf* and its use in modifying the threshold for low rate blowing snow (equation 4.5.1). Added parameter *asr_probability* to the product and text in section 4.6.2.3.

For this version, 8.2, **dated August 21, 2019**, relative humidity profiles were added to the ATL04 and carried forward to the ATL09 product (Table 3.1). Changes were made to the numerical values in Table 4.4 which defines the *bsnow_con* parameter. Background method 1 was changed to better calculate the background in twilight conditions. Also, the background computation was changed in the area affected by the South Atlantic Anomaly (SAA). Made the alpha constant (in equation 3.15) in the computation of molecular folding a function of beam and solar elevation angle (pages 33-34). Removed section 3.5 – removal of ground return from the cloud layer array – the layer finding code, detailed in ATBD Part II, has been changed so that this is no longer necessary. The value of *phi_land* and *phi_ocean* have been changed to 1.05 and 0.85, respectively and have been added to the ATL09 product. The value of the adjustable parameter *cal_solar_elev_max* has been changed from 0.0 to 3.0. The parameters *layer_conf_dens*, *asr_cloud_probability* and *bsnow_intensity* have been added to the ATL09 product (Table 4.1). Changed the bin to start the search for blowing snow from *surface_bin-1* to *surface_bin-2*. Clarified the wording in the discussion of multiple scattering warning flag in section 4.4. Section 3.3.4.4 – Theoretical Background added that the result of equation 3.25 must also be multiplied by 400.0 to account for the 400 shot sum.

For this version 8.1, **dated March 12, 2019**, corrections to the calibration constants of calibration method 2 were made (section 3.3.7.2) and changes were made to section 3.5, ground removal from cloud layer array. The changes are in red font as are the changes made in version 8.0.

For this version 8.0, **dated February 21, 2019**, background method 1 algorithm (section 3.3.4) was totally revised. Two new calibration methods were added, so that now there are 3 separate calibration methods (section 3.3.7.2). Changes to the surface signal retrieval (section 3.3.5) were made. The computation of the cloud fold flag (old ATL09 parameter *Cloud_fold_flag*) was moved from L3a to L2a. This parameter will now reside on ATL04 and carried forward to the ATL09 product (previously it was only on ATL09). The algorithm for the computation of *Cloud_fold_flag* was also augmented. These changes are in new sections 3.4 and 3.4.1 which

together replace old section 4.4.2, which was deleted. New parameters associated with the above changes were added to the ATL04 product parameter list. The adjustable parameter *Calib_select* was added to the ATL09 product parameter list. Section 3.5 was added to document the removal of the ground return from the cloud layer array as defined by the DDA algorithm. All changes are in red font. Note that the background and calibration changes were handed in to the project in early January, 2019 during the government shutdown.

For version 7.5, **dated June 16, 2018** changed the calibration solar angle limit from zenith to elevation (section 3.3.7.2) and in Table 3.6. Changed the wording of the TEP removal in section 2.3. Modified equation 4.7 (apparent surface reflectance) by replacing the energy received with the number of photons received and removed equation 4.8 and associated text as it is no longer necessary. Re-made figure 4.6.1 using new modified equation 4.7. Modified the molecular scattering folding equation (eqn. 3.15) to replace the number of photons transmitted with laser energy. Added section 3.3.4.1 the computation of the theoretical background (equation 3.25). Added a new parameter (*backg_theoret*) to the ATL09 product defined in section 3.3.4.1. All relevant changes in red.

For version 7.4, **dated April 10, 2018**, all changes in red font. We have modified the apparent surface reflectance equation 4.7 by replacing T_{opt} with S_{ret} which is defined as the system receiver response and comes from ATL03.

The deadtime equation 2.1 had an error. Please replace with new equation 2.1.

The equation to compute molecular folding (equation 3.15) was altered. The product of the detector quantum efficiency (Q_e) and the receiver optical transmission (T_{opt}) was replaced with S_{ret} .

Section 2.3 was modified to correct the equations for removal of the Transmit Echo Pulse (TEP) from the atmospheric profiles.

In section 4.4, some clarification wording was added to explain the parameter *msw_flag*.

Added section 6.9, multiple scattering warning flag plot.

For **version 7.3, dated November 15, 2017**, all changes in red font. We have added a paragraph to section 2.1 – Dead Time Correction – which introduces a control parameter “*dtime_select*” which is used to determine which dead time factor to use in the processing.

Added some wording to section 2.1 –Dead Time Correction for clarification.

Changed equation 2.1 to now define just the dead time correction factor and added equation 2.1a to define how the dead time correction factor is applied.

Added “*dtime_select*” parameter to Table 3.5

Deleted two sentences from the first paragraph of section 3.3.5 to avoid confusion.

Added discussion of how to apply dead time correction to section 3.3.5

Added a paragraph describing added parameters to the ATL09 product in Section 4.2. These include density_pass2 and renaming old “density” to density_pass1. Updated Table 4.1 in that section to reflect this change.

Added parameter Surface_Bin to ATL09 output in Table 4.1. This parameter is stored on ATL04 and should simply be passed to ATL09.

Fixed Table 4.2 description of “Surface_Signal_Source” to be consistent with text.

Rewrote section 4.5 – Blowing Snow. Added Table 4.3 on page 55.

In Section 4.6.2.1, updated the input GOME data set for surface reflectance from a 1x1 degree to the new 0.25x0.25 degree data set. Also changed the value of R_{snow} to 0.8 from 0.6.

For version 7.2, **dated July 28, 2017**, note all changes in red font.

Updated section 2.1 – dead time correction.

Added Surface_Signal_Source (Table 4.2) control parameter to define the source of the surface signal in computation of Apparent Surface Reflectivity. Paragraph describing this is in section 4.6, page 56.

Modified surface signal computation – step 4 – in section 3.3.5, page 37.

Added ATL09 parameter *Surface_Signal_Source* to Tables 4.1 and 4.2. This is a constant read in from an ancillary file that tells whether to use the surface signal from ATL03 or that from ATL04 to compute apparent surface reflectance.

Changed section 4.4 to specify a maximum of 10 cloud layers (was 6).

Added definition of *layer_attr* flag to section 4.4.

Added clarification to section 4.5 – Blowing Snow.

Added new section 5.0 – Consolidated Cloud Flag. Old sections 5 and 6 become new sections 6 and 7. Added new parameter *layer_flag* to ATL09 (Table 4.1).

Outstanding issues: The TEP correction must be implemented and tested. The quality assurance parameters (Section 6) have not yet been added to the products.

For Version 7.1 **dated March 28, 2017**, changes and additions to the prior version (Version 7.0, dated October 20, 2016) have been entered in red font. The main changes are:

Added Section 2 – Corrections to the Raw Atmospheric Profiles. This section addresses dead time correction, vertical shifting of profiles and the transmit echo pulse (TEP) correction. Prior version dead time correction was in section 2.3.5.1. This version dead time correction procedure has not changed from the prior version and is complete except for identifying the laboratory-produced detector efficiency table. Discussion of vertical shifting (section 2.2) and the TEP (section 2.3) are new to this version.

In Table 3.1 – List of ATL04 product parameters – added parameter *Backg_Method3* which is the background computed from the 200 Hz onboard background. This is defined in section 3.3.4 on page 32. Also added in Table 3.1 is the ATL04 parameter *mol_backs_folded*. Though the procedure to compute this parameter was in the prior ATBD version, it was not in the ATL04 product parameter list.

Added further explanation of the application of the dead time correction factor in Section 3.3.5

In Table 4.1 – List of ATL09 Parameters – added parameter *Backg_Select* (from ATL04) and parameter *Bsnow_Prob* which is defined in section 4.5.1

For Version V7.0, dated **October 22, 2016**, changes and additions to the prior version (Version 6.0, dated November 3, 2015) have been entered in red font. The main changes are:

Replaced section 2 with new section entitled “Correction to the Raw Atmospheric Profiles”. The previous ATBD version section 2 becomes section 3, section 3 becomes 4, etc.

Added more complete discussion of dead time correction in section 2.1 including Figure 2.2.

Section 2.3: Added discussion of the removal of the Transmit Pulse Echo (TPE) from the atmospheric profiles

Secant theta added to equation 3.12

Added more MET parameters to ATL04 and ATL09 output

Added two dead time correction parameters (Dtime_Fac1 and Dtime_Fac2) to ATL04 and ATL09

Added ATL03 background and surface signal parameters to ATL04 and ATL09

Added folded molecular profile to ATL09 product (Equation 3.19a)

Added some pseudo code to vertical alignment section (section 3.3.6)

Discussion added to section 3.3.4 on the use of the onboard 50 shot background

Multiple scattering warning flag (Msw_flag) added to ATL09 product

Section 4.4 discussion of cloud flags added

Changed the ATLAS receiver transmission from 0.30 to 0.49 in Table 1.1

Energy of strong beams now 120 μ J in Table 1.1

Added discussion of calibration in Section 4.3

Added Cal_Default ANC parameter to Table 4.2

Added section 4.4.2 and equation 4.5a for the computation of a new ATL09 parameter called Cloud_fold_flag

Added section 5.0 – Quality Assessment

Added section 6.0 – Product Quality Parameters

Added Appendix B – Geopotential height to geometric height conversion

The main changes for version V6.0, dated **November 3, 2015** from the prior version (V5.0 dated June, 2015) are:

Section 2.1, new required input to ATL04 (ATL03)

Added section 2.3.2.1

Updated section 2.3.5.1 – Dead time correction

Revised section 3.5 and added Table 3.4

Revised section 3.6.2 – ASR Cloud detection algorithm implementation

Updated Tables 2.7 and 3.1

Version V5.0 was the first to split the ATBD into two parts. Part I addresses L2A (ATL04) and all of L3A (GLA09) except for layer detection which is handled exclusively in a separate document: ICESat-2 Algorithm Theoretical Basis Document for the Atmosphere, Part II: Detection of Atmospheric Layers and Surface Using a Density Dimension Algorithm, V5.0 dated **4 June 2015**.

The main differences to this document (Part I) compared to the prior version (V4.0 dated 1 November, 2014) are listed below:

- 1) Addition of section 2.3.2 – Molecular Scattering Folding Correction. This attempts to remove the molecular scattering folded from above from the measured signal.
- 2) Modified background computation detailed in Method 2.
- 3) Added section 2.3.6.3 – Calibration Algorithm using Surface Reflectance
- 4) Corrected errors and added clarification to the ozone transmission calculation.
- 5) Fixed Equation 2.7
- 6) Changed equation 2.10 to be consistent with CALIPSO ATBD
- 7) Updated Table 3.1
- 8) Put code for vertically interpolating pressure into Appendix A

1 Introduction and Background

Scheduled for launch in 2017, the ICESat-2 satellite will carry only one instrument – ATLAS (Advanced Topographic Laser Altimeter System) which utilizes a high repetition rate (10 KHz) 532 nm laser and photon counting detectors for high resolution altimetry measurements. It will use a diffractive optical element (DOE) to produce 6 individual laser beams simultaneously emitted from the satellite. Three of the beams will have nominal energies of about 30 μ J (weak beams) and the other 3 will have energies roughly 4 times the weak beams (strong beams). The altimetry measurements will utilize all 6 laser beams while the atmospheric measurements will only use the 3 strong beams. Each footprint of the strong beams will be separated by about 3 km on the ground (across track) as shown in Figure 1.1.

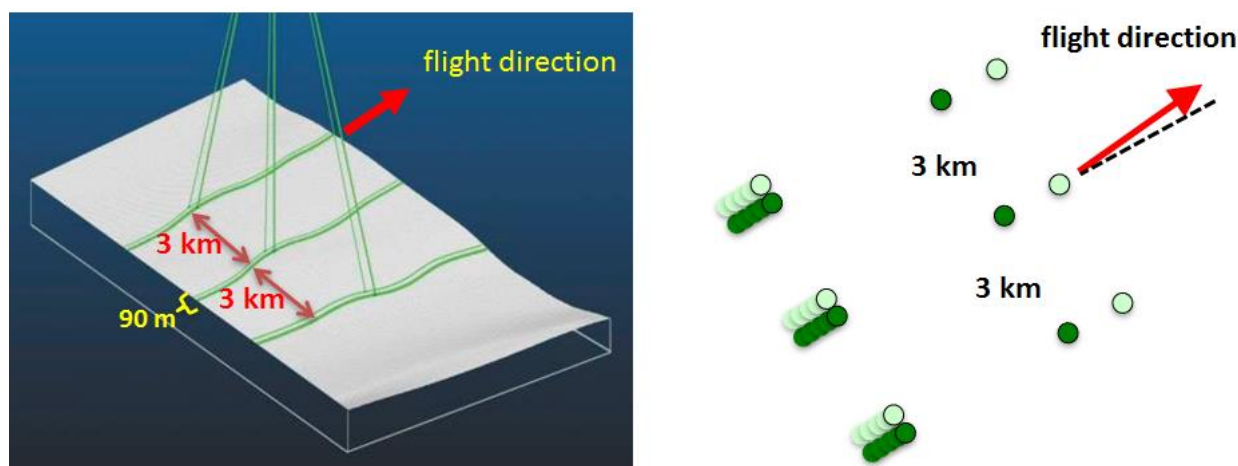


Figure 1.1 ICESat-2 laser beams and surface tracks. The satellite is yawed by 2 degrees such that the weak beams trail the strong beams and vice versa depending on spacecraft orientation which is determined by solar beta angle. Separation of strong and weak beam tracks is 90 m on the ground. Beam pairs are separated by 3 km.

The ICESat-2 atmospheric profiles will consist of 30 m bins in a 14 km long column. Nominally the top of the column will be 13.75 km (above the local value of the onboard Digital Elevation Model (DEM)) and the bottom -0.250 km. For the atmosphere, the 3 strong beams (approximately 120 μ J at 532 nm) will be downlinked after summing 400 shots, resulting in three 25 Hz profiles (280m along track resolution). Thus, each summed, 25 Hz profile is equivalent to roughly 48 mJ of energy, which is about twice the level of GLAS 532 L2A measurements. However, GLAS used geiger mode Single Photon Counting Module (SPCM) detectors which have a very low dark count rate and high quantum efficiency (60%). ATLAS will use Photo Multiplier Tube (PMT) detectors which are inherently noisier (with a factor of 10 to 100 more dark count rate compared to SPCMs) and less efficient. At present, the best estimate of dark count rate for the ATLAS detectors is between 10 and 50 KHz. The exact nighttime performance of the atmospheric channel will depend on the magnitude of the dark count rate. This is

illustrated in Figure 1.2 which compares the expected nighttime performance of ATLAS for two different values of the detector dark count rate.

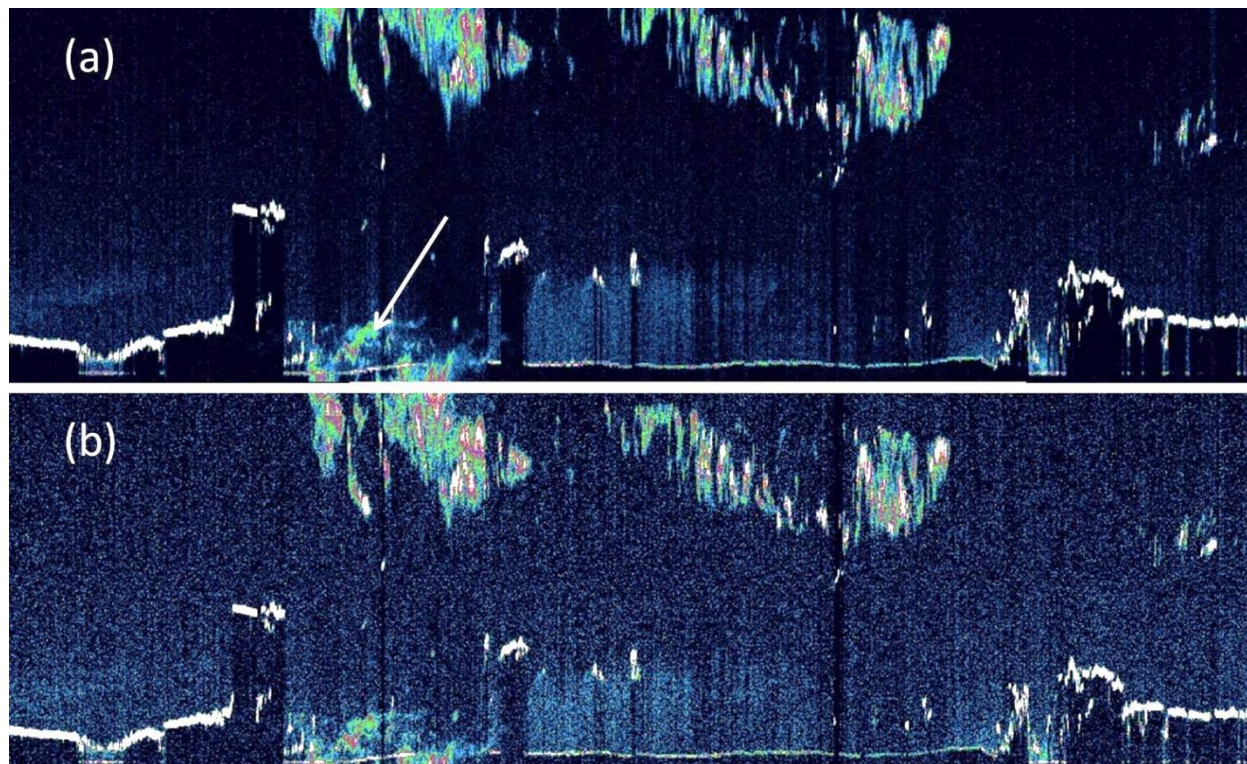


Figure 1.2. Full resolution (25 Hz, 400 shot sum) simulation of the nighttime performance of the ATLAS atmospheric channels (3 strong beams). The top panel (a) is for a detector dark count rate of 10 KHz and the bottom panel (b) is 5 times that number. The actual on-orbit detector dark count rate will likely lie somewhere in between these extremes. The scattering at the position of the arrow in (a) is from a cloud above 15 km (refer to aliasing discussion below).

Figure 1.2 shows simulated nighttime ATLAS data at full resolution (25 Hz or 400 shot sum). The top panel is for a dark count rate of 10 KHz, while in the bottom panel the dark count rate is five times that (50 KHz). Laboratory measurements to characterize the ATLAS instrument indicate initial dark count rate to be about 10 KHz. However this value will increase over time on orbit. As can be seen, the nighttime performance of ATLAS is expected to be very good, with both thin cloud and aerosol easily detected even for the higher dark count rate.

For daytime, the magnitude of the detector dark count does not matter as much since the performance will be determined by the solar background which, on average, will be between 1 and 5 MHz (about 100 times the detector dark count rate). Daytime ATLAS data will be noisier than the GLAS L2A data because when summing signals over some fixed time interval the solar noise in a lidar system scales with the laser repetition rate (GLAS was 40 Hz; ATLAS is 10KHz). This degradation in daytime signal to noise (compared to GLAS) is likely to be a factor of 100 or more. Thus, during the day thin cloud and aerosol will be detectable only after a large

amount of averaging, if at all. Model results like those shown in Figure 1.3 indicate that in order to detect thin cloud and aerosol during daytime would require averaging at least 100,000 lidar pulses (10 seconds or about 70 km horizontal resolution).

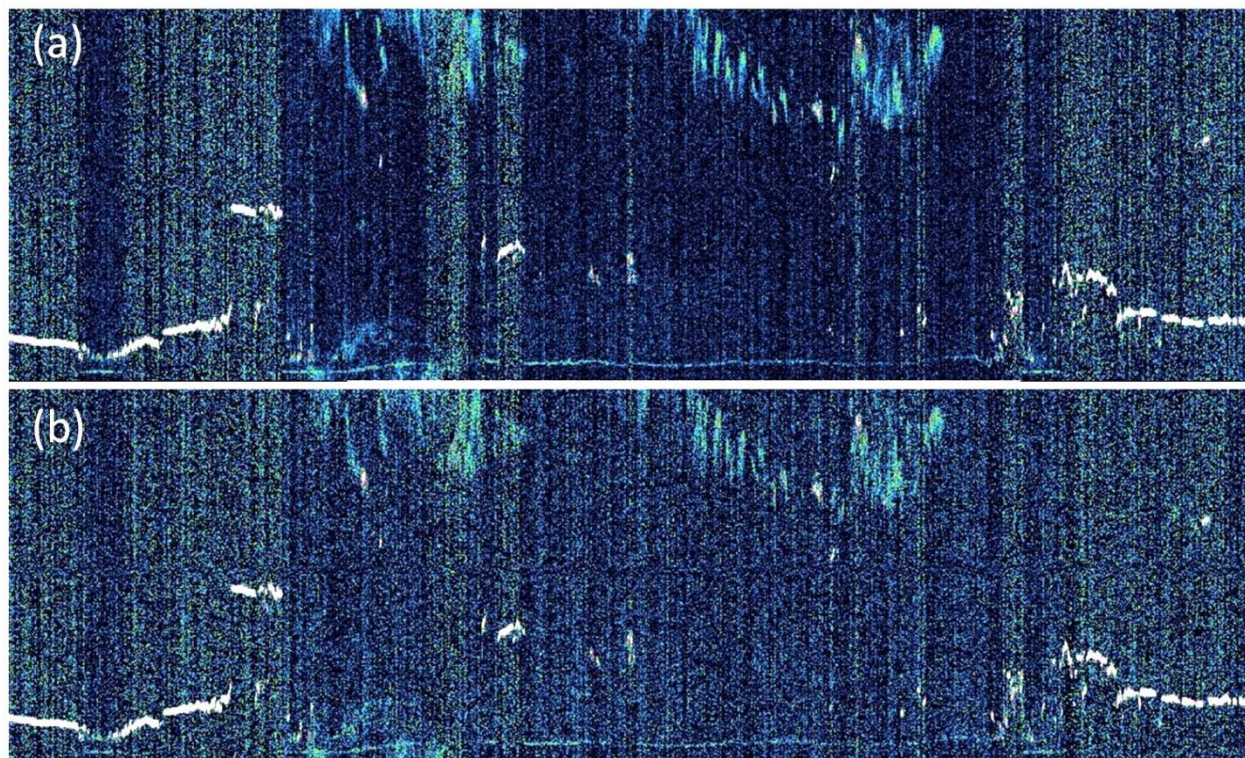


Figure 1.3. Simulation of the ATLAS atmospheric channel (1 beam) based on GLAS 532 nm data. Shown is the sum of 400 shots (full ATLAS resolution for the atmosphere) for daytime over a surface albedo of 0.10 (a), and daytime over a surface with albedo 0.30 (b).

Table 1.1 Pertinent ATLAS instrument parameters and their values at present.

ATLAS Instrument Parameter	Current (pre-lab measured) Value
Laser Repetition Rate	10 KHz
Laser Energy (strong beams)	120 μ J
Telescope Effective Area	0.43 m ²
Telescope FOV	83 μ r
Detector Quantum Efficiency	0.15
Detector Dead Time	3 ns
Detector Dark Count Rate	10-100 KHz
Bandpass Filter Width	30 pm
Nominal Receiver Optics Throughput	0.49
Nominal Orbit Height	495 km
Laser/Telescope FOV Spot Size (on ground)	14 m/41 m

While the daytime background presents a big problem for ATLAS, there are other problems that are even more formidable. To acquire high resolution altimetry measurements, ATLAS is using a high repetition rate laser (10 KHz). Each laser pulse will be separated by only 30 km in the vertical. Thus, when a pulse (pulse 1) strikes the ground, the laser pulse right after it (pulse 2) is at 30 km altitude. When the ground return from pulse 1 reaches 15 km altitude (on its way back to the satellite), laser pulse 2 is at 15 km also (but travelling downward). The atmospheric return from pulse 2 (from 15 km altitude) will travel back to the receiver at the same time as the ground return from pulse 1. Thus, if there were a cloud at 15 km, its scattering signature would appear at the position of the ground return. Clouds at 16 km altitude will appear at 1 km above the ground, etc. This effect will be termed “pulse aliasing” or “folding” in future references within this document. To put it in a more general way, the atmospheric scattering that will be recorded by the instrument at height H is the sum of the scattering at height H , $H+15$, $H-15$, $H+30$, $H-30$, $H+45$, $H-45$ etc., where the numbers are in km. Even in the tropics, 99.9% of clouds are below 18 km. Thus, there will be a range in the downlinked profiles from -0.25 km to about 3 km that can potentially contain cloud scattering from 14.75 to 18 km. Above 3 km, this effect should be minimal (with exception of the highest of clouds in the tropics and Polar Stratospheric Clouds). Pulse aliasing can be seen in the simulated data of Figure 1.1a at the position of the arrow. The scattering shown there is actually from the top of the cloud above 15 km. Above 3 km to the top of the profile (nominally 13.75 km), we should have no ambiguity in the scattering signatures, since there is usually only molecular scattering above 18 km (of course this does not apply to Polar Stratospheric Clouds that typically occur in the altitude range 12-25 km over the polar region in winter, or stratospheric aerosol).

Even though ATLAS can theoretically capture 15 km of data, the instrument will only download 14 km of data. Unfortunately the ATLAS instrument will require a hardware reset between shots that will effectively create a blind spot in the top 1 km of the profile (nominally 14-15 km). In this region no information is returned. Thus, if a cloud exists in this region we will never know it except for the case where it is optically thick and has the effect of attenuating the laser beam such that no other cloud or surface return is seen below it.

These “characteristics” produce severe problems for lidar data processing. First, there is nowhere in the profile where one can calculate the solar background. Normally, this is done using the signal either very high up (> 100 km as in the case of GLAS) or below the ground. However, we have no signal either very high up or below ground, since for instance, the scattering in the roughly 500 m of the profile below ground will be equal to the sum of the scattering at 14.5 – 15.0 km, 29.5 – 30.0 km, etc. So even if there were no particulate scattering at 14.5-15 km, the signal below ground will still have the molecular return in the 15 to 14.5 km altitude range. This can be modeled and subtracted from the computed background when there are no clouds. If clouds are present (in the 15 to 14.5 km altitude range) then there would be no way to recover the background from the 0.50 km segment of data below ground. Yet another problem caused by the high repetition rate laser pertains to calibration. With a normal (lower repetition rate)

atmospheric lidar, calibration is performed using the molecular return from a known, clear region of the atmosphere normally in the low to mid stratosphere (25-35 km). For instance for GLAS, we used the 28 to 24 km altitude range for calibration. CALIPSO uses the 36-39 km region. ATLAS will not have access to this region used by GLAS (or more correctly, it will be folded into the scattering between 6 and 9 km altitude). Calibration also requires long stretches (100's of km) of data devoid of particulate scatterers. For ICESat-2, the only place that might offer the necessary conditions for calibration is over the poles during darkness or very low sun angle where there is a high probability that no clouds will exist between 13.5 and 11-12 km. Darkness or at least twilight is required because the signal from molecular scattering (i.e. the calibration target) is so small, that it would be buried in the solar background noise for daytime data. However, in the polar regions in winter, polar stratospheric clouds may exist and will contaminate the calibration if not recognized and removed from the calibration procedure.

The unique characteristics of ICESat-2 atmospheric data discussed above represent unprecedented difficulties in producing level 2 and 3 atmospheric products. It should be stressed, however, that ATLAS is not intended to be an atmospheric lidar. ICESat-2 is not an atmospheric mission. Any atmospheric information that can be obtained is a bonus to the project, but its quality will undoubtedly be less than ideal for atmospheric science. The main intent of the atmospheric channel is the detection of clouds, blowing snow and fog that will adversely affect the altimetry measurements. The mission objective (for the atmospheric data) is to produce a cloud/no cloud flag and to loosely characterize the likely intensity of multiple scattering of the pulse due to clouds, fog and blowing snow. This information is highly valuable to the altimetry analysis. This objective can be met without accurate calibration of the data because we have designed a layer finding algorithm that does not require calibrated data to find clouds and blowing snow. Thus in summary, it will be challenging to produce an atmospheric product but the main objective of aiding the altimetry data analysis is attainable. We will, however, have to come up with creative ways to get around the above cited problems if we are to have a useful atmospheric product.

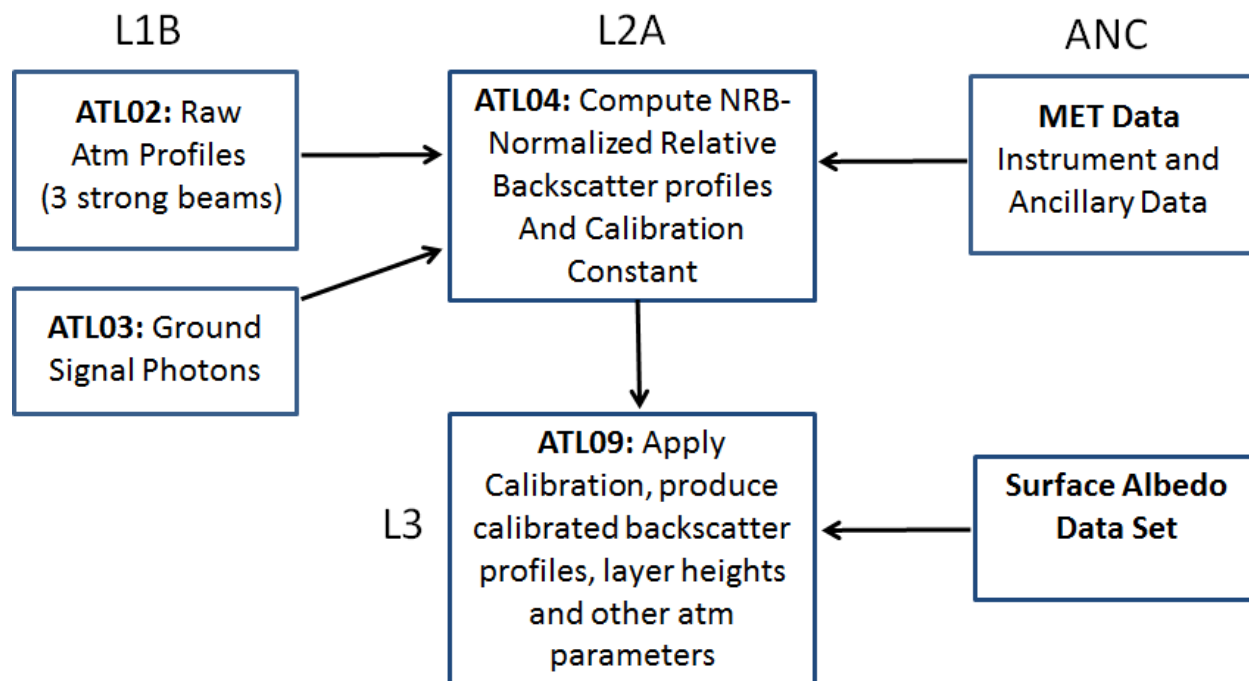


Figure 1.4. Overview of processing steps from the raw atmosphere profiles (ATL02) to the L3 atmosphere product (ATL09).

2 Corrections to the Raw Atmospheric Profiles

There are a number of corrections that need to be applied to the raw atmospheric profiles before they are processed to produce the level 2 products. These include removal of the transmit echo pulse (TEP), dead time correction and compensation for shifting between major frames. These issues are discussed below. Before we address those issues, we should note that the 3 strong beams which make up the atmosphere profiles will be named by convention atmospheric profile 1, profile 2 and profile 3. These three profiles correspond to ATLAS beam 1, 2 and 5. In the initial spacecraft orientation, ATLAS beam 1 is to the left of the nadir ground track, ATLAS beam 2 is along the nadir track and beam 5 is to the right. The spacecraft flies in two orientations, depending on the solar beta angle. The spacecraft rotates 180 degrees from one orientation to the other. Thus, sometimes beam 1 is on the left side of the ground track and sometimes on the right. Likewise with beam 5. ATLAS beam 2 is always in the center regardless of the spacecraft orientation. There will be a parameter called `sc_orient` that will tell the user which orientation the spacecraft is in. When `sc_orient` = 1 (spacecraft facing forward), ATLAS beam 1 will be to the right of the nadir ground track and will map to atmospheric profile 3. When `sc_orient` = 0, (spacecraft facing backward), ATLAS beam 1 is on the left side of the nadir ground track and will map to atmospheric profile 1. For both spacecraft orientations, ATLAS beam 2 will map to atmospheric profile 2.

2.1 Dead Time Correction

Unlike the SPCM detectors of the GLAS 532 nm channel, the ATLAS detectors will have much shorter dead times (about a factor of 10 less). In addition, the low laser pulse energy of ATLAS (about a factor of 200 less than GLAS) means that detector dead time will not be important for atmospheric signals, as even dense clouds will not produce more than 3 or 4 photons per 30 m bin per shot. However, this does not apply to surface signals that can produce 10 to 20 photons per shot and, for some surfaces, may arrive in just a few nanoseconds. Dead time correction of surface signals is thus very important. Figure 2.1 shows a model of the ATLAS detector efficiency (labeled lines) as a function of the received return pulse strength (y axis) and width (x axis).

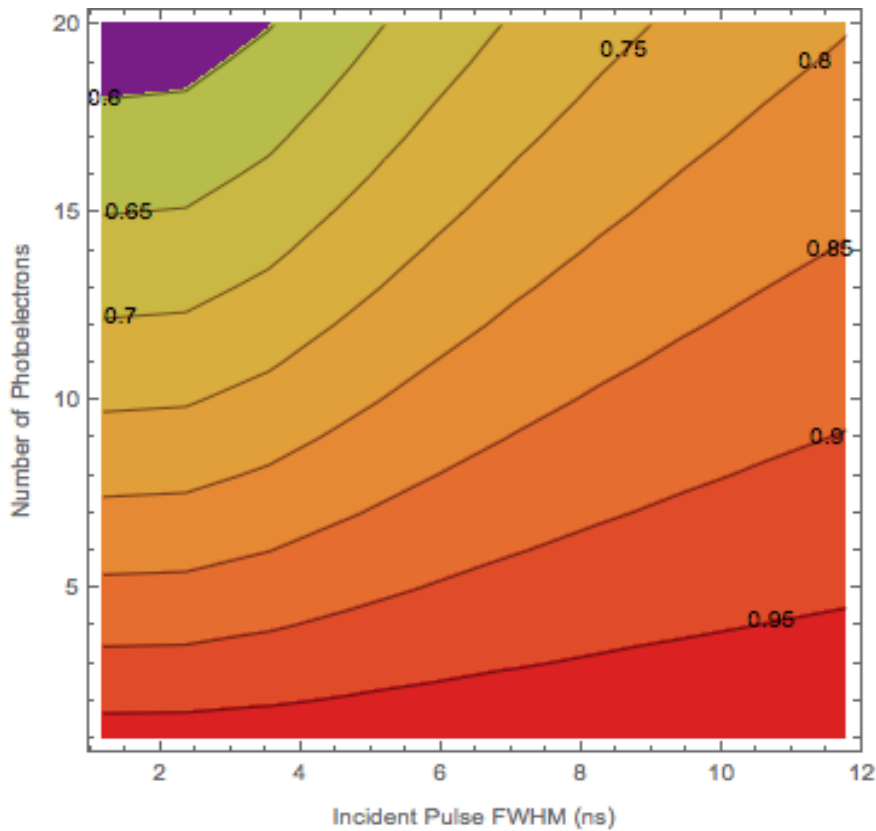


Figure 2.1 A model of effective radiometric efficiency as a function of actual (incident) return strength (photoelectrons per shot) and actual surface return pulse width (in nanoseconds).

As can be seen from the plot, the stronger and shorter the return pulse, the less efficient the detector is at capturing all the photons. This loss is due to the dead time effect. The figure also shows that in order to correct for this effect, we have to know both the return signal strength and the width of the return pulse. The latter is problematic as it is not recorded by the instrument but can be obtained by the aggregation of many shots. Tables which relate the dead time correction factor to the ground signal strength and width have been produced as part of the calibrations done on ATLAS in the laboratory (CAL 34). The table y axis contains the number of photons received for a single shot and the x axis is the width of the pulse in nanoseconds. There are 6 tables in all, one for each value of detector dead time ranging from 2.9 ns to 3.4 ns. There are 6 of these tables which are for specific detector dead times:

CAL_34_DT29_2017172.csv
 CAL_34_DT30_2017172.csv
 CAL_34_DT31_2017172.csv
 CAL_34_DT32_2017172.csv
 CAL_34_DT33_2017172.csv

Where “DT29” means dead time 2.9 ns, “DT30” means dead time 3.0 ns, etc. Thus, we must have a procedure to choose which CAL 34 table to use.

The dead time for each of the three strong beam atmospheric profiles are computed by the average of each of their detectors 1-16 dead times. The dead times for each of these 16 detectors are contained in CAL 42 tables which depend on temperature. The correct CAL 42 table will be identified in the current granule ANC39 ancillary file. We need to average the 16 deadtimes contained in the CAL 42 file to identify which CAL 34 table to use for the dead time correction. We will use the CAL 34 table with the deadtime closest to the averaged value. Once the appropriate CAL 34 table has been identified, it is then used to obtain the dead time correction factor by the procedure shown in Figure 2.2 below. This procedure uses parameters from ATL03 to define the surface signal strength (*sig_count_hi*, *sig_count_med* and *sig_count_low*). These parameters are read in by ATL04 and represent the surface signal photon count summed over 400 shots reported by confidence level (high, medium and low). Also from ATL03 are the parameters *sig_h_sdev_hi*, *sig_h_sdev_med* and *sig_h_sdev_low* which are the standard deviation of the height of all signal (surface) photons for the 400 shot interval. Figure 2.2 below shows the logic we propose to use for computing the dead time correction using the ATL03 parameters and the radiometric table (CAL 34) discussed above. All six of these parameters in addition to *sig_h_mean_hi*, *sig_h_mean_med* and *sig_h_mean_low* are to be stored on the ATL04 product (see Table 3.1).

We propose to sum *sig_count_hi* and *sig_count_med* and divide by 400 to obtain the signal strength value in photons per shot (y axis on Figure 2.1). To compute the signal width, if *sig_count_hi* is greater than 200, we will use *sig_h_sdev_hi* multiplied by a factor (Dead_Time_Sfac, **for this ATBD version 8.5, set to 0.5**) to represent the signal width and convert to nanoseconds (divide by half the speed of light). If *sig_h_sdev_hi* is invalid, then set *Dtime_fac1* to invalid.

If *sig_count_hi* is less than or equal to 200 and *sig_count_med* is not invalid, we compute the sum of *sig_count_hi* and *sig_count_med*. If this sum is less than or equal to 200, we set the dead time correction factor to 1.0. If this sum is greater than 200, then we compute the signal width by summing *sig_h_sdev_hi* and *sig_h_sdev_med*, multiplying by Dead_Time_Sfac and dividing by the speed of light. If *sig_h_sdev_med* is invalid, then only use *sig_h_sdev_hi* multiplied by Dead_Time_Sfac divided by half the speed of light to compute the signal width. If *sig_h_sdev_hi* is invalid, then set *Dtime_fac1* to invalid. The resulting signal width and signal strength are used to index into the laboratory-produced detector efficiency table to obtain the dead time correction factor (*Dtime_Fac1*). Note that the dead time correction factor obtained from CAL 34 will be a number ≥ 1.0 . To correct the *Surface _signal* parameter for dead time, it should be multiplied by *Dtime_fac1*.

NOTE: If the above data needed to compute *Dtime_Fac1* are not available, then its value should be set to invalid. See section 3.3.5 on how to obtain *Surface_signal*.

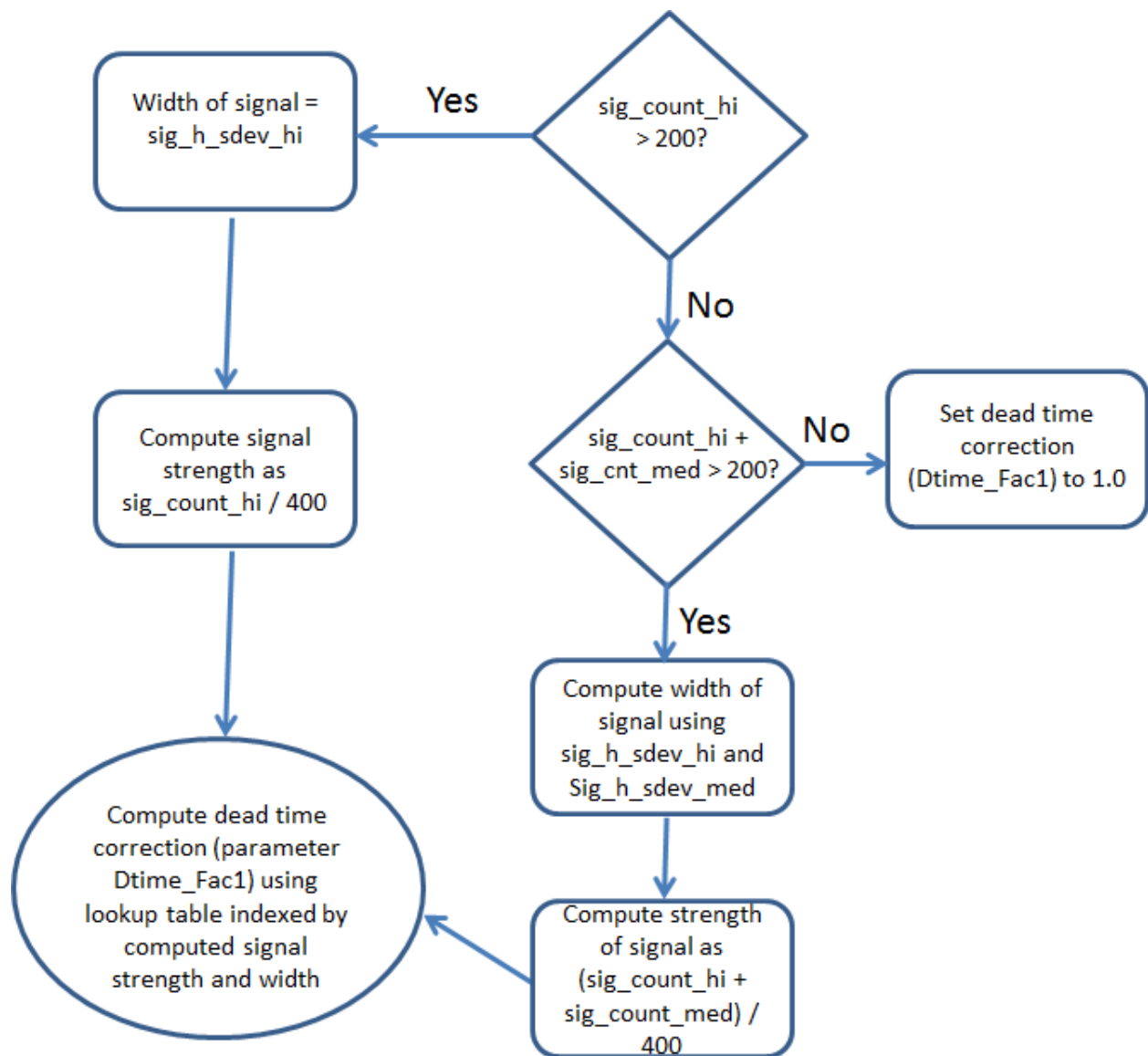


Figure 2.2. Flow diagram for computing the dead time correction factor using ATL03 measured surface signal photon count and standard deviation.

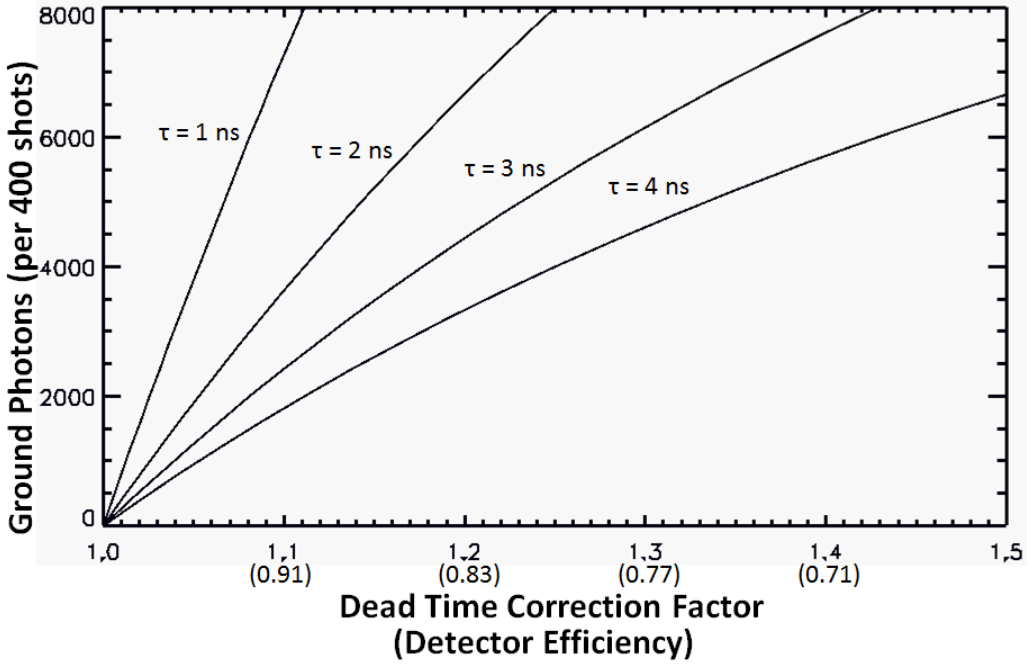


Figure 2.3. Detector dead time correction factor and efficiency as computed from equation 2.1 as a function of received photons (400 shot sum) within a 30 m bin for different values of detector dead time (τ).

The dead time correction can also be computed analytically. This method does not include the effects of pulse width and is thus not as accurate as the above method.

$$D_c = \frac{1.0}{1.0 - \frac{\tau P_m}{80.0 \times 10^{-6}}} \quad (2.1)$$

$$P_c = D_c P_m \quad (2.1a)$$

Where D_c is the dead time correction factor (*Dtime_fac2*), P_m is the measured photon count (surface signal, per bin per 400 shots), τ is the detector dead time ($\sim 3.0 \times 10^{-9}$ s), and P_c is the dead time corrected signal. The exact value of τ to use can be calculated from the appropriate CAL42 table by averaging the dead time for all 16 strong spot detectors. The atmosphere bins are 30 m which is equivalent to 0.20 microseconds, so the dead time must be scaled to this time interval. To do so we divide the detector dead time by the bin length (in time – 0.20×10^{-6} s) multiplied by the number of shots summed (400). The latter accounts for the fact that the ATLAS atmosphere data are sums of 400 shots. Figure 2.3 shows the result of equation 2.1. Referring to Figure 2.1, the effect of roughness is to broaden the return pulse, which decreases the effect of dead time. It is as if increasing surface roughness decreases the dead time of the detector (see the 4 curves in Figure 2.3). This value (D_c from equation 2.1) should be written to

ATL04 as parameter *Dtime_Fac2* in addition to *Dtime_Fac1*. P_m in equation 2.1 is in fact the surface signal (parameter *Surface_Sig*) as determined in section 3.3.5. Thus to calculate *dtime_fac2*, one must first have the magnitude of the surface signal in photons per bin. See section 3.3.5 for the method to compute the surface signal (parameter *Surface_Sig*).

~~Note that if no surface signal was found, then *Dtime_Fac1* and *Dtime_Fac2* should be set to invalid.~~

Equation 2.1 defines *dtime_fac2*. It is a number between 1.0 and 2.0. Usually about 1.1 – 1.5. The surface signal only (*Surface_Sig*) is multiplied by that factor to dead time correct the surface signal. No need to apply this to atmospheric bins. Equation 2.1 is an approximation to the method used to compute *dtime_fac1*. We want to define a control parameter that specifies which dead time factor to use. Call this "dtime_select". If 1, then use *dtime_fac1*, if 2 then use *dtime_fac2*. Default value of 1. The code should use what "dtime_select" specifies, but if it is asking for *dtime_fac1* and for some reason, it cannot be calculated, it should be smart enough to default to *dtime_fac2*. *Dtime_fac2* can always be calculated as long as there is a surface signal. *Dtime_select* should be on both ATL04 and ATL09. **Note: for ATBD 8.5 and version 005 data products, dtime_select=2).**

2.2 Correction for Raw Profile Shifting

The raw atmosphere profiles are constructed from two sets of 200 shot summed profiles onboard the satellite. The top bin of all 200 shots of each sum will begin at the same height above the WGS84 ellipsoid. It is possible that the height of the top bin of the next 200 shot sum is different than that of the previous 200 shot sum. In this case, when the flight software sums the two 200 shot sums together to produce the 400 shot sum that is downlinked, the shifting can result in some bins of the 400 shot profile containing only the sum of 200 shots. The flight software removes the bins at the top of the profile with only a 200 shot sum and the bins with only a 200 shot sum at the bottom of the profile will be retained. To determine how many bins at the end of the profile do not overlap (and, hence, contain the original 200 shot values), the flight software computes the parameter *atm_shift_amount* that is stored on the ATL02 product. For those bins (with only a 200 shot sum) the ATL04 processing is to double the photon counts in the lower bins therein to make them consistent with the rest of the 400 shot sum profile. The bins to be doubled are simply the last "*atm_shift_amount*" bins of each 25 Hz profile. This correction is required for each of the 3 profiles. Note: There is a slight difference in the start of the range window, to the order of about 40 ns. This currently is not corrected for in the L2A processing.

2.3 Transmit Echo Pulse

The ATLAS beams 1 and 2 will have within them the transmit echo pulse (TEP). The position of the TEP can be calculated and should reside within 1 or 2 atmospheric bins (30 m wide) for each 400 shot (summed) profile. We will need to calculate the location of this bin and replace it with a Poisson distributed value based on the magnitude of the average of the two bins above and below the position of the TEP. The algorithm to compute the location of the TEP within the atmospheric profile is given below. This correction must be made to ATLAS beams 1 and 2 (atmospheric profiles 1 and 2). ATLAS beam 5 (atmospheric profile 3) will not contain the TEP. This correction should be the first thing done before any other ATL04 processing.

To compute the TEP location within the atmospheric profile, we need the range from spacecraft to the top of the atmospheric profile in nanoseconds. Call this value R. Then the time of flight (TOF) of the TEP center position (in nanoseconds) is:

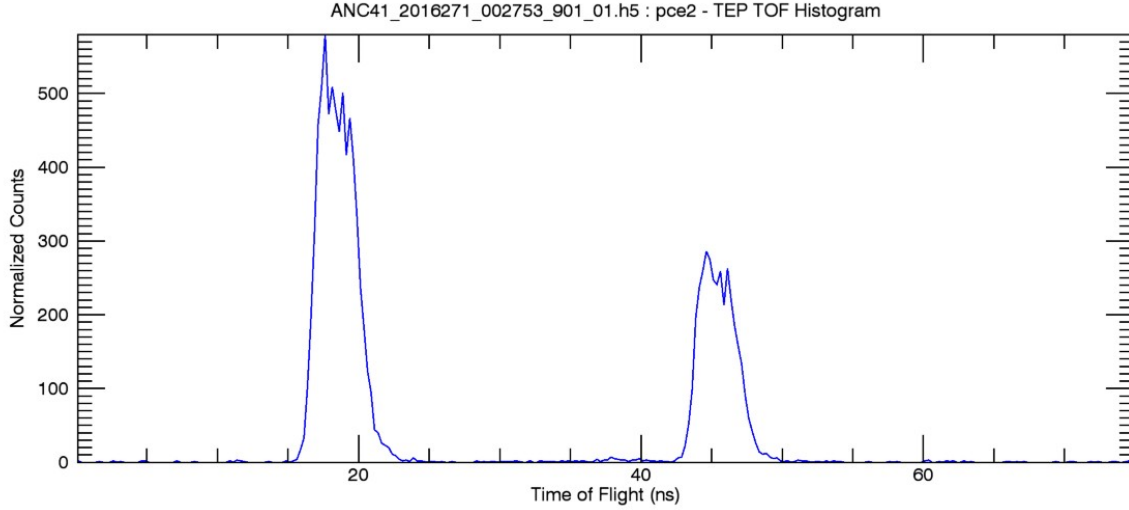


Figure 2.4 The Transmit Echo Pulse (TEP) shape and magnitude as a function of time from laser fire. The smaller second pulse (past 40 ns) is the TEP echo.

$$T = \text{INT}(R/\text{prf_inv}+1)*\text{prf_inv}+\text{TEPstart}+0.5*\text{TEPwidth} \quad (2.2)$$

Where TEPstart is a parameter that varies by picoseconds but can be considered constant for our purposes. TEPstart is a parameter on ATL03 (/ancillary_data/tep/tep_range_prim) and should be put on ANC39 and used in equation 2.2. The nominal value of TEPstart is 15 ns. TEPwidth will also be a constant and will equal the time between the beginning of the TEP and the end of the TEP echo as seen in Figure 2.4. Thus, the nominal value for TEPwidth is 40 ns. Prf_inv is the time in seconds between laser pulses (= 1.0e-4s). The bin of the atmospheric profile that contains the TEP is then:

$$\text{TEP_bin} = (T-R)/\text{ATMbinwidth} \quad (2.2a)$$

Where ATMbinwidth is the atmospheric bin width in seconds (= 200 e-9s). This bin and the bins immediately above and below it (as a margin of safety) must be replaced by the adjacent atmospheric bins as:

```
first = .TRUE.
mu = (atm_bins(tep_bin-1) + atm_bins(tep_bin+1)) / 2.0
tep_val = random_Poisson(mu,first)
atm_bins(tep_bin) = tep_val
```

Where ATM_bins represents the atmospheric profile.

3 L2A Product: Normalized Relative Backscatter (ATL04)

The atmosphere L2A product will be called ATL04 and will consist of what is termed Normalized Relative Backscatter (NRB) profiles and calculated 532 nm calibration coefficients (plus other ancillary/supporting data parameters). The NRB profiles are created from the profiles of raw photon counts (supplied from ATL02 – parameter /atlas/pcex/atmosphere_sw/atm_bins) by subtracting the background, multiplying by the square of the range from the satellite to the return height and normalizing by the laser energy. It was recognized by the reviewer of this document that since scattering within a bin originates from multiple altitudes, there is no unique range that can be applied to the bin. We now recognize this and have devised a method which will help alleviate this effect. In section 3.3.2 we develop a method to remove the molecular scattering from adjacent laser pulses from the recorded profile. We recognize that this method cannot remove particulate scattering that has been folded down into the profile, but it is the best that can be done.

The calibration coefficient (C) is derived from the NRB profiles (and knowledge of the temperature and pressure of the atmosphere) and is used to compute calibrated, attenuated backscatter profiles (which will be on the ATL09 product, not on ATL04). We do not know how variable C will be or how it will change with time. Normally, the factors that cause C to change are 1) boresite misalignment, 2) detector changes in responsivity, 3) changes in laser characteristics not accounted for by laser energy monitor data, and 4) overall changes in either or both the transmit and receive (optics) system throughput. Past experience with GLAS and CALIPSO indicates that calibration changes are mainly related to temperature change effects on the boresite as the satellite travels from sunlight to darkness and vice versa. However, since ATLAS will have an active continuous boresite alignment process, this may not be the case for ICESat-2. A thorough discussion of the calibration coefficient and how it is calculated is presented in section 3.3.7. There is also the possibility of using the transmit echo, which will be recorded onboard periodically, to monitor changes in the receiver optics transmission and detector quantum efficiency. These two parameters are part of what determines the value of C for lidar systems (see Equation 3.26).

For the computation of latitude and longitude (of the laser footprint on the surface), the range from the satellite to the topmost bin of the atmosphere profile will suffice for surface (laser) spot location determination. The top of the profile will be the determined height above the WGS84 ELLIPSOID for this geolocation.

3.1 L2A Required Inputs

Raw strong beam atmosphere profiles (ATL02 parameter atm_bins)

ATL03: Ground signal photon magnitude and width

Laboratory data relating signal magnitude and width to detector efficiency (see Figure 2.1).

Meteorological data: wind, temperature, pressure, relative humidity and others (GMAO)

Ozone concentration (mixing ratio), source GMAO

DEM at 1x1 km resolution (GMTED2010 or as defined in ATL03 ATBD)

Surface Type (IGBP)

Solar azimuth and zenith angle

Range from Spacecraft to start of atmosphere profile (ATL02 parameter atm_rw_start)

Spacecraft altitude (wrt the WGS84 ellipsoid)

Pointing angles for each beam

Laser energy for each of the 3 strong beams, time, lat/lon

Onboard 50 shot background (ATL02)

200 shot sum shift amount - ATL02 parameter atm_shift_amount

The raw atmosphere profiles are the sum of 400 shots from the 3 strong beams and are assumed to come from ATL02 no with dead time correction applied.. The meteorological data should be the initial analysis fields (or a short term forecast less than 12 hour) from a global weather prediction model such as GEOS-5 (from GMAO) or the GFS or NAM models (from NCEP). The ozone concentration is included in the GMAO files.

3.2 L2A Outputs

- Normalized Relative Backscatter (NRB) profiles for each of the 3 strong beams at 25 Hz.
- Background at 25 Hz (includes Bin_{max1} , Bin_{max2} , S_{m1} , S_{m2} , S_{std} and p_{b1} and S_{tot} , P_{mtot} and p_{b2} – see section 3.3.2 below for definitions)
- Molecular backscatter profile from 20 to 0 km at 0.1 Hz, 30 m vertical resolution.
- Calibration coefficients with time and location tags
- Molecular Backscatter Average used to compute C (calibration coefficient)
- Signal average used to compute C
- Pressure, temperature from 30 to 0 km at 0.1 Hz, 30 m vertical resolution.
- Range from spacecraft to start of atm profile
- Pointing angles for each beam
- Laser energy, time, lat/lon
- Solar zenith and azimuth angle, DEM, surface type
- Surface (2 m) wind velocity and temperature
- Wind velocity and temperature at 10 m height
- Onboard 50 shot background
- MET data defined in Table 3.1

Table 3.1. ATL04 Product Parameters

Parameter	Datatype	Units	Rate	Description
Time	float	seconds	25 Hz	
Latitude	double	degrees	25 Hz	Based on range to local DEM
Longitude	double	degrees	25 Hz	
prof_dist_x	float	m	25 Hz	Along track distance from start of segment
prof_dist_y	float	m	25 Hz	Across track distance from reference pair track
Bckgrd_counts	Integer	photons	200 Hz	Onboard 50 shot background counts

Bckgrd_counts_reduced	Integer	photons	200 Hz	Onboard 50 shot background counts minus signal photons
Bckgrd_rate	Float	Photons/s	200 Hz	50 shot background rate in per second (signal photons removed)
Bckgrd_hist_top	Float	m	200 Hz	50 shot background integration window top ht
Bckgrd_int_height	Float	m	200 Hz	50 shot background integration window width
Bckgrd_int_height_reduced	Float	m	200 Hz	bckgrd_int_height – signal photon height range
Sig_count_hi	Integer	photons	25 Hz	Ground signal photons for a 400 shot sum, high confidence
Sig_count_med	Integer	photons	25 Hz	Ground signal photons for a 400 shot sum, med confidence
Sig_count_low	Integer	photons	25 Hz	Ground signal photons for a 400 shot sum, low confidence
Sig_h_mean_hi	Float		25 Hz	Mean height of surface wrt EGM2008 ellipsoid, high confidence
Sig_h_mean_med	Float		25 Hz	Mean height of surface wrt EGM2008 ellipsoid, medium confidence
Sig_h_mean_low	Float		25 Hz	Mean height of surface wrt EGM2008 ellipsoid, low confidence
Sig_h_sdev_hi	Float		25 Hz	Standard deviation of surface height, high confidence
Sig_h_sdev_med	Float		25 Hz	Standard deviation of surface height, medium confidence
Sig_h_sdev_low	Float		25 Hz	Standard deviation of surface height, low confidence
NRB_Prof	Float(700, 3)	Photons m ² /Joule	25 Hz	3 strong beams from 20 to -1 km (based on local DEM value) with vertical resolution of 30 m
NRB_Top_Bin	Integer(3)	NA	25 Hz	The starting (top) bin number within the -1 to 20 km frame where data begins
NRB_Bot_Bin	Integer(3)	NA	25 Hz	The ending (bottom) bin number within the -1 to 20 km frame where data ends
Backg_Method1	Float(3)	photons/bin	25 Hz	Background from method 1
Backg_Std_Dev2	Float(3)	photons/bin	25 Hz	Background standard deviation from method 12 (S_{std2} defined in section 3.3.4)
Backg_Mean2	Float(3)	photons/bin	25 Hz	Signal mean from method 2 (S_{bar} defined in section 3.3.4)
Backg_Method2	Float(3)	photons/bin	25 Hz	Background from method 2 (p_{b2} defined in section 3.3.4)
Backg_Method3	Float(3)	photons/bin	25 Hz	Background from method 3 (defined in section 3.3.4)
Backg_Select	Integer	NA	1/granule	The background method used in calculation of NRB
backg_night_scale_factor	Float	NA	1/granule	Scaling factor for nighttime background (method 1)
backg_day_scale_factor1	Float	NA	1/granule	scaling factor for daytime background (method 1)
backg_day_scale_factor2	Float	NA	1/granule	scaling factor for daytime background (method 1)

backg_min_solar_elev	Float	Degrees	1/granule	Maximum Solar elevation angle for data to be classified as nighttime (method 1)
backg_max_solar_elev	Float	Degrees	1/granule	Minimum solar elevation angle for data to be classified as daytime (method 1)
backg_night_background	Float	Photons/Bin	1/granule	Value to be used for background when data is classified as nighttime (method 1)
backg_twilight_scale_factor	Float	NA	1/granule	Scaling factor to be applied in calculation of background in twilight conditions (method 1)
backg_nseg	Integer	NA	1/granule	Number of segments to break the atm histogram into for computation of daytime background (method 1)
backg_day_exp_factor	Float	NA	1/granule	Factor applied to daytime background (method 1)
backg_response_time	Float	Seconds	1/granule	Used for appx dead time correction of background (method 1)
saa_latmin	Float	Degrees	1/granule	Latitude minimum of box that encompasses the area affected by the South Atlantic Anomaly (SAA) (method 1)
saa_latmax	Float	Degrees	1/granule	Latitude maximum of box that encompasses the area affected by the South Atlantic Anomaly (SAA) (method 1)
saa_lonmin	Float	Degrees	1/granule	Longitude minimum of box that encompasses the area affected by the South Atlantic Anomaly (SAA) (method 1)
saa_lonmax	Float	Degrees	1/granule	Longitude maximum of box that encompasses the area affected by the South Atlantic Anomaly (SAA) (method 1)
saa_scale_fac	Float	NA	1/granule	Scale factor for computing the background in method 1 within the south Atlantic anomaly box
Cal_N	Integer	NA	1/granule	Number of calibration points for this granule
Cal_C	Float(3, Cal_N)	Photons*m ³ /sr/J	Cal_N/orbit	Calculated calibration coefficients (one per beam, eqn 3.27)
Cal_Con	Integer(3, Cal_N)	NA	Cal_N/orbit	Calibration Confidence
Cal_Start_Time	float(Cal_N)	UTC	Cal_N/orbit	Start Times of calibration calculation
Cal_End_Time	float(Cal_N)	UTC	Cal_N/orbit	End Times of calibration calculation
Cal_Start_Latitude	float(Cal_N)	degrees	Cal_N/orbit	Start Latitude of calibrations
Cal_Stop_Latitude	float(Cal_N)	degrees	Cal_N/orbit	Stop Latitude of calibrations
Cal_Start_Longitude	float(Cal_N)	degrees	Cal_N/orbit	Start Longitude of calibrations
Cal_Stop_Longitude	float(Cal_N)	degrees	Cal_N/orbit	Stop Longitude of calibrations
Cal_Molec	float(Cal_N)	m ⁻¹ sr ⁻¹	Cal_N/orbit	Molecular Backscatter value used to compute Cal_C

Cal_NRB	float(Cal_N)	NA	Cal_N/orbit	NRB value used to compute Cal_C
Cal_C_Trans	float(Cal_N)	NA	Cal_N/orbit	The total transmission used to compute Cal_C
Cal_Ozone_Trans	float(Cal_N)	NA	Cal_N/orbit	Ozone transmission term used to compute C
Calib_Select	integer	NA	1/granule	Used to select the calibration method. 1=cal method 1, 2=cal method2, 3=cal method 3
Calib_night_pce1	Float	Photons*m ³ sr/J	1/granule	Calibration constant for nighttime data for pce1 atm profiles for calibration method 2
Calib_night_pce2	Float	Photons*m ³ sr/J	1/granule	Calibration constant for nighttime data for pce2 atm profiles for calibration method 2
Calib_night_pce3	Float	Photons*m ³ sr/J	1/granule	Calibration constant for nighttime data for pce3 atm profiles for calibration method 2
Calib_day_pce1	Float	Photons*m ³ sr/J	1/granule	Calibration constant for daytime data for pce1 atm profiles for calibration method 2
Calib_day_pce2	Float	Photons*m ³ sr/J	1/granule	Calibration constant for daytime data for pce2 atm profiles for calibration method 2
Calib_day_pce3	Float	Photons*m ³ sr/J	1/granule	Calibration constant for daytime data for pce3 atm profiles for calibration method 2
Calib_twilight_pce1	Float	Photons*m ³ sr/J	1/granule	Calibration constant for twilight data for pce1 atm profiles for calibration method 2
Calib_twilight_pce2	Float	Photons*m ³ sr/J	1/granule	Calibration constant for twilight data for pce2 atm profiles for calibration method 2
Calib_twilight_pce3	Float	Photons*m ³ sr/J	1/granule	Calibration constant for twilight data for pce3 atm profiles for calibration method 2
calib_solar_elev_min	Float	degrees	1/granule	Solar elevation angle less than which the data are considered nighttime for calibration methods 2 and 3
calib_solar_elev_max	Float	degrees	1/granule	Solar elevation angle greater than which the data are considered daytime for calibration methods 2 and 3
nrb_smooth	Integer	NA	1/granule	Number of points to average the NRB data for calibration method 3
default_nrb_day	Float(3)	Photons km ² /Joule	1/granule	Default value for daytime NRB – used to replace NRB values that are out of range for each pce for calibration method 3
default_nrb_night	Float(3)	Photons m ² /Joule	1/granule	Default value for nighttime NRB - used to replace NRB values that are out of range for each pce for calibration method 3
min_nrb_day	Float(3)	Photons m ² /Joule	1/granule	Minimum daytime NRB accepted for filtered NRB data for each pce for calibration method 3
max_nrb_day	Float(3)	Photons km ² /Joule	1/granule	Maximum daytime NRB accepted for filtered NRB data for each pce for calibration method 3
min_nrb_night	Float(3)	Photons m ² /Joule	1/granule	Minimum nighttime NRB accepted for filtered NRB data for each pce for calibration method 3
max_nrb_night	Float(3)	Photons m ² /Joule	1/granule	Maximum nighttime NRB accepted for filtered NRB array for each pce for calibration method 3

nrb_average_period	Integer	NA	1/granule	Number of seconds to average the smoothed and filtered NRB array before computing the calibration constant for calibration method 3
min_calib_day	Float(3)	Photons*m ³ sr/J	1/granule	Minimum calculated calibration allowed for daytime for each pce for calibration method 3
max_calib_day	Float(3)	Photons*m ³ sr/J	1/granule	Maximum calculated calibration value allowed for daytime for each pce for calibration method 3
min_calib_night	Float(3)	Photons*m ³ sr/J	1/granule	Minimum calculated calibration value allowed for nighttime for each pce for calibration method 3
max_calib_night	Float(3)	Photons*m ³ sr/J	1/granule	Maximum calculated calibration value allowed for nighttime for each pce for calibration method 3
MET_slp	Float	Pa	1Hz	Sea Level Pressure
MET_Latitude	Float	deg	1 Hz	Latitude of MET data
MET_Longitude	Float	deg	1 Hz	Longitude of MET data
MET_PS	Float	Pa	1 Hz	Surface Pressure
MET_U2M	Float	ms ⁻¹	1 Hz	Eastward component of wind at 2m height
MET_U10M	Float	ms ⁻¹	1 Hz	Eastward component of wind at 10m height
MET_U50M	Float	ms ⁻¹	1 Hz	Eastward component of wind at 50m height
MET_V2M	Float	ms ⁻¹	1 Hz	Northward component of wind at 2m height
MET_V10M	Float	ms ⁻¹	1 Hz	Northward component of wind at 10m height
MET_V50M	Float	ms ⁻¹	1 Hz	Northward component of wind at 50m height
MET_T2M	Float	K	1Hz	Temperature at 2 m height
MET_T10M	Float	K	1Hz	Temperature at 10 m height
MET_QV2M	Float	Kg/Kg	1 Hz	Specific humidity at 2 m height
MET_QV10M	Float	Kg/Kg	1 Hz	Specific humidity at 10 m height
MET_TS	Float	K	1 Hz	Surface (skin) temperature
MET_TROPPB	Float	Pa	1 Hz	Blended tropopause pressure
MET_TROPT	Float	K	1 Hz	Tropopause temperature
MET_TQL	Float	Kg/m ²	1 Hz	Total column cloud liquid water
MET_TQI	Float	Kg/m ²	1 Hz	Total column cloud ice
MET_CLDPRS	Float	Pa	1 Hz	Cloud top Pressure
Mol_Backscatter	Float(700)	m ⁻¹ sr ⁻¹	1 Hz	Molecular backscatter profile, 30 m resolution, -1 to 20 km
Molec_bkscat_p	Float(700)	PA	1 Hz	Atmospheric pressure profile, 30 m resolution, -1 to 20 km
Molec_bkscat_t	Float(700)	K	1 Hz	Atmospheric temperature profile, 30 m resolution, -1 to 20 km
Molec_bkscat_rh	Float(700)	%	1 Hz	Atmospheric relative humidity profile, 30 m resolution, -1 to 20 km
Molec_Trans	Float(700)		1 Hz	Molecular transmission profile, 30 m resolution, -1 to 20 km
Ozone_Trans	Float(700)		1 Hz	Ozone transmission profile, 30 m resolution, -1 to 20 km
Mol_Backs_folded	Float(700)	m ⁻¹ sr ⁻¹	1 Hz	Folded molecular backscatter profile (equation 3.17)

Dtime_select		NA		Control parameter for determining which method to use to compute the dead time correction factor
Dtime_Fac2	Float(3)	NA	25 Hz	Dead time correction factor for surface signal computed from equation 2.1
Dtime_Fac1	Float(3)	NA	25 Hz	Dead time correction factor for surface signal computed from radiometric lookup table
Surface_Signal	Float(3)	Photons	25 Hz	Surface signal from the 3 strong beams
Surface_Bin	Integer(3)	NA	25 Hz	Bin number of surface return
Surface_Height	Float(3)	m	25 Hz	Surface height (from signal) for each beam
Surface_Thresh	Float(3)	Photons	25 Hz	Computed signal threshold for surface detection for each of the 3 beams
Surface_Width	Integer(3)	Bins	25 Hz	The number of bins comprising the surface signal for each beam. If > 1, the surface bin and height correspond to highest of those bins and the parameter Surface_Signal is sum of the signal value of those bins.
Surface_Conf	Float(3)	NA	25 Hz	Confidence value for the surface detection. Ranges from 1.0 (lowest confidence) to 100.0 (highest confidence).
Grnd_thresh1_sfac	Float	NA	1/granule	Adjustable constant for computation of Surface Threshold
Grnd_thresh2_sfac	Float	NA	1/granule	Adjustable constant for computation of Surface Threshold
Grnd_search_width	Integer	bins	1/granule	Width in bins of the search window for ground detection
Cloud_fold_flag	Byte (3)		25 Hz	Flag to tell user that this profile likely contains cloud signal folded down from above 15 km to the last 2-3 km of the profile. One flag for each profile. See Table 3.9 for flag value meanings
Fold_nbins	Integer	NA	1/granule	Number of bins starting from end of raw profile to compute mean of signal to test for presence of cloud folding
Fold_thresh_night	Float	Photons/bin	1/granule	Raw signal level above which the mean of the last “fold_nbins” indicates the presence of cloud folding for nighttime data
Fold_thresh_day	Float	Photons/bin	1/granule	Raw signal level above which the mean of the last “fold_nbins” indicates the presence of cloud folding for daytime data
Range_S	float	km	25 Hz	Range from spacecraft to top of profile
ATLAS_Altitude	Float	m	25 Hz	Altitude of Spacecraft wrt the WGS84 ellipsoid
ATLAS_PA	float	degrees	25 Hz	Pointing angle of spacecraft
Laser_Energy	Float(3)	Joules	10 Hz	Laser energy of each beam
Solar_Zenith	float	degrees	1 Hz	Solar Zenith Angle
Solar_Azimuth	float	degrees	1 Hz	Solar Azimuth Angle
Dem_h	float	km	25 Hz	DEM value from a 1 kmx1km
Surf_type_igbp	integer	NA	25 Hz	IGBP Surface Type
Podppd_flag	byte	NA	25 Hz	Precision orbit and pointing flag from ATL03

Note: Data Product quality parameters not listed

3.3 NRB Computation

The first step in the lidar data processing is to compute what we call normalized relative backscatter (NRB) from the raw level 0 data. In this step, three corrections to the data are made: 1) Laser energy normalization, 2) range square correction and 3) background subtraction. The lidar equation is:

$$S(z) = \frac{CE\beta(z)T^2(z)}{r^2} + p_b + p_d \quad (3.1)$$

In the above, r is the range from the spacecraft to the height z , $S(z)$ is the measured raw signal (photons) at height z , C is the lidar system calibration coefficient, E the laser energy, $\beta(z)$ the 180° backscatter coefficient at height z , $T(z)$ the one way atmospheric transmission from the spacecraft to height z , p_b the solar background and p_d the detector dark count rate. The NRB (computed for each of the 3 strong beams) is then:

$$NRB(z) = (S(z) - p_b - p_d)r^2/E = C\beta(z)T^2(z) \quad (3.2)$$

For ICESat-2, the most difficult thing in the above is the solar background (p_b) computation as both E and r are well known. In practice we will lump p_b and p_d together and their sum will be called simply ‘background’. At night, with no moon (and no effect from city lights), the background will simply be the dark count rate (p_d), while during the day the background will be a sum of p_b and p_d . The NRB will constitute an L2A product (Parameter *NRB_Prof* on ATL04). ICESat-2 is using PMT detectors with very short dead times (3 ns). The count rates associated with atmospheric scattering and solar background will not require dead time correction (unlike the SPCMs used in GLAS), however, dead time is important for the ground return. The ground return signal (in photon counts) will be computed in ATL04 and carried forward to ATL09 where it is used to compute Apparent Surface Reflectance (ASR) which is a level 3 product (ATL09) and is discussed in section 4.

Prior to computing the NRB, the raw photon count profiles ($S(z)$) need to be corrected for the folding of molecular scattering from above 15 km into the recorded profile. This correction is discussed in section 3.3.2, after we discuss how to compute molecular backscatter below.

3.3.1 Molecular Backscatter Computation

A fundamental parameter that will be used in a number of places in this document is the molecular backscatter coefficient (β_m). Molecular backscatter is a function of atmospheric density which is computed from the ancillary MET data. We will compute β_m from 60 km altitude to the ground once per second along the orbit track. Though only 20 km will be stored on the ATL04 product, we need to compute the molecular backscatter up to 60 km to correct the raw signal for molecular scattering folded into the profile from above (see section 3.3.2). Also on the product will be the latitude and longitude of the molecular backscatter profile (ATL04 parameters *MET_Latitude* and *MET_Longitude*).

The MET data (as obtained from the GMAO ancillary data) are reported at standard pressure levels which include temperature, relative humidity and the geopotential height at that pressure level. The geopotential height is very close to the geometric height (above mean sea level) and only varies by a few percent due to gravitational variation (mostly) as a function of latitude. Please see Appendix B for how to convert from geopotential height to geometric height. The pressure ($P(z)$), temperature ($T(z)$) and relative humidity ($RH(z)$) are calculated for the bins (heights) between the standard pressure levels. The temperature and relative humidity can be linearly interpolated (to 30 m vertical resolution) from the model pressure levels. The pressure cannot be (linearly) interpolated and must be computed using the hypsometric formula (Byers, 1974):

$$z_2 - z_1 = \frac{RT}{g} \ln(P(z_1)/P(z_2)) \quad (3.3)$$

In equation 3.3, let us set

$$\frac{RT}{g} = \psi$$

Where \bar{T} is the average temperature of the layer between z_1 and z_2 , g the gravitational constant and R the ideal gas constant for dry air (see Table 3.2). Now we re-write equation 3.3 as:

$$\psi = \frac{(z_{m2} - z_{m1})}{\ln(P(z_{m1})/P(z_{m2}))} \quad (3.4)$$

Here, z_{m1} and z_{m2} are the heights of the model temperature and pressure data and $z_{m2} > z_{m1}$. The top model height will ideally be greater than 60.0 km as we are computing the pressure (and following that, the molecular backscatter) to 60 km altitude.

Then, for all z between z_{m1} and z_{m2} we compute the pressure at lidar bin resolution ($\Delta z=30$ m here) as:

$$P(b_i) = P(b_{i-1}) \exp \left[\frac{(\Delta z)}{\psi} \right] \quad (3.5)$$

Where b_i denotes the lidar bin number. Here we are computing the pressure at 30 m increments starting at the lowest model level (highest pressure) and working upward in height. Thus Δz will always be 30 m (in this case).

An example program, written in IDL is shown in Appendix A. This program is used with radiosonde data as a demonstration, but it can just as well be used with model data such as GEOS-5. Note that this code will produce profiles of pressure, temperature and molecular backscatter from the ground upward (bin 0 at the ground). In the ATL04 product we want bin 0

to be at the top of the profile. Also, relative humidity is not included in the code, but if available in the model data, it can be linearly interpolated as is done for the temperature. The moisture effect on the calculation of molecular backscatter is negligible except for possibly within the boundary layer.

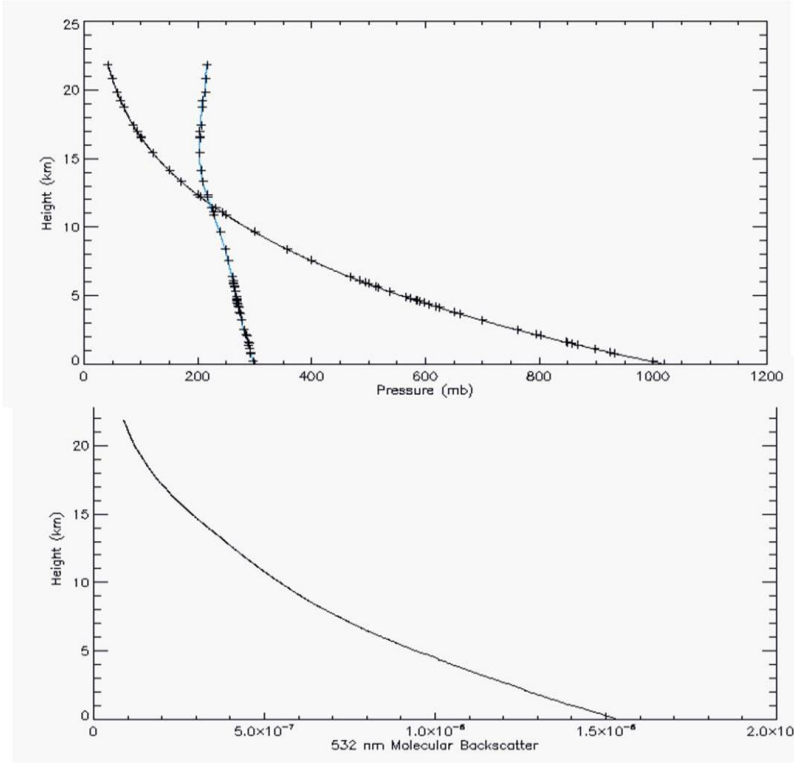


Figure 3.1 Output from the code in Appendix A to interpolate pressure and temperature (blue line, °K) from observation levels to 30 m lidar bin resolution (top). Molecular backscatter calculated from the resulting temperature and pressure profiles using equations 3.6 and 3.10 (virtual temperature was not used here).

The output of the code in the Appendix is shown above in Figure 3.1 above. The ‘+’ signs indicate the radiosonde data for both the pressure (black line) and the temperature (blue line).

From the calculated temperature, relative humidity and pressure profiles, the molecular number density ($N(z)$) is calculated from the ideal gas law as:

$$N(z) = P(z) / (k T_v(z)) \quad (3.6)$$

where $N(z)$ is in units of molecules per cubic centimeter, k is the Boltzmann constant for dry air in units of ergs per degree per molecule ($k=1.3806488 \times 10^{-16}$ ergs per degree Kelvin), P is the atmospheric pressure in units of ergs per cm^2 (or millibars times 10^3), and T_v is the virtual temperature in degrees Kelvin. This equation is very similar to the equation to compute atmospheric density ($\rho(z)$), which is the same as (3.6) except that the Boltzmann constant is

replaced by the ideal gas constant for dry air (R), which has a value of $287.058 \text{ m}^2 \text{ s}^{-2} \text{ }^\circ\text{K}^{-1}$. Note that we will be computing atmospheric density for the computation of ozone transmission, in equation 3.21. The effect of moisture on atmospheric density is included through the use of the virtual temperature in equation 3.6, but these effects are generally negligible above the lower troposphere. T_v is computed from the relative humidity (obtained from the MET data) by first converting it to water vapor mixing ratio. To accomplish this, we need to first compute the saturation vapor pressure (e_s) which is a function of the atmospheric temperature (T (in Kelvin)) as:

$$e_s = 0.6112 e^{(17.67(T-273.16))/(T-29.66)} \quad (3.7)$$

and from that compute the saturation mixing ratio (q_s):

$$q_s = 0.622 e_s / (P / 10.0) \quad (3.8)$$

where P is the atmospheric pressure in millibars. The relative humidity is simply the actual atmospheric water vapor mixing ratio divided by the saturation mixing ratio times 100. Thus, the actual atmospheric water vapor mixing ratio is given by

$$q = r q_s / 100.0$$

where r is the relative humidity. And finally, the formula to compute the virtual temperature (T_v) is:

$$T_v = \frac{T}{1.0 - 3q/5} \quad (3.9)$$

Following She (2001) and the CALIPSO ATBD, from the atmospheric molecular number profile, the molecular backscatter coefficient ($\beta_m(z, \lambda)$) in units of $\text{m}^{-1} \text{sr}^{-1}$ is then:

$$\beta_m(z, \lambda) = 5.1909 N(z) (550.0 / \lambda)^4 10^{-26} \quad (3.10)$$

where λ is the wavelength in nanometers (532 in our case). The molecular backscatter profile (parameter *Mol_Backscatter*) will be computed up to 20 km altitude and will be on the ATL04 product at 1 Hz. Also on the product at that rate will be the temperature and pressure profiles (parameters *MET_Temp* and *MET_Pres*) at 30 m vertical resolution.

Table 3.2 Constants Used in this Section

Constant	Value
Boltzman (k)	$1.3806488 \times 10^{-16}$ ergs per degree Kelvin

Ideal Gas Constant (R)	287.058 m ² s ⁻² °K ⁻¹
Gravitational constant at sea level (g)	9.80171 ms ⁻²

3.3.1.1 Molecular Transmission

To calculate the molecular transmission, $T_m(z)$, we first compute the molecular extinction profile ($\sigma_m(z)$), by multiplying the molecular backscatter cross section by the molecular extinction to backscatter ratio, which is known theoretically to be $8\pi K_f/3$.

$$\sigma_m(z) = 8\pi K_f \beta_m(z) / 3 \quad (3.11)$$

Where K_f is known as the King factor (Bodhaine et al., 1999) which has a value of 1.0401. The molecular optical thickness from the top of the profile (ztop, here we are using 60 km for ztop) to height z is equal to the integral of the molecular extinction profile as shown in equation 2.12

$$\tau_m(z) = \text{Sec}(\theta) \int_{ztop}^z \sigma_m(z) dz \quad (3.12)$$

Where θ is the off nadir pointing angle. Finally, the two-way molecular transmission ($T_m^2(z)$) between ztop and any height z is:

$$T_m^2(z) = e^{-2\tau_m(z)} \quad (3.13)$$

The attenuated molecular backscatter is defined as:

$$\beta_m(z) T_m^2(z) T_o^2(z) \quad (3.14)$$

Where $T_o^2(z)$ is the ozone transmission from the top of the atmosphere to height z (see equation 3.23 and 3.24 in section 3.3.3).

3.3.2 Molecular Scattering Folding Correction

As discussed in sections above, the raw photon data captured by ATLAS at height z (where z is ≤ 14 km) will have contributions from atmospheric scattering at height z+15 km, z+30 km, z+45 km, etc. It is important to remove as much of this as possible. There will be particulate and molecular scattering contributions, but we will have no knowledge of the former. The best that can be done is to model and remove the latter. The molecular contribution to the received photon count can be computed from equation 3.15:

$$P_m(z) = \frac{E}{r^2} \beta_m(z) \Delta z A_t T_m^2(z) T_o^2(z) S_{ret} N_a R(z) \alpha \quad (3.15)$$

In equation 3.15, α is used to adjust the computed photon count in case equation 3.15 is obviously producing too many or too few photons. Initially, for versions 001 and 002 α had a constant value of 0.0 for all profiles. After launch, analysis of data has shown that alpha is a function of solar angle and profile number. Table 3.2a below lists the values alpha should be

given for day, night and twilight. Day, night and twilight are defined as they are for calibration method 2: day = solar elevation \geq cal_solar_elev_max; twilight = solar_elevation $<$ cal_solar_elev_max and solar_elevation \geq cal_solar_elev_min; night = solar_elevation $<$ cal_solar_elev_min. The values of cal_solar_elev_max and cal_solar_elev_min are defined in section 3.3.7.2.

Table 3.2a

Light Environment	PCE Number	Alpha
Night	3	4.0
Night	2	4.2
Night	1	4.7
Day	3	-4.0
Day	2	-5.0
Day	1	-4.0
Twilight	3	-0.1
Twilight	2	-0.5
Twilight	1	1.5

The other terms used in equation 3.15 above are:

E – The laser energy in Joules

r – The range from the satellite to the height z (in m).

$\beta_m(z)$ – the molecular backscatter cross section at height z ($\text{m}^{-1} \text{sr}^{-1}$). See section 3.3.1 and equation 3.10.

Δz – the bin size in meters (30 m)

A_t – Area of telescope (m, effective)

$T_m(z)$ – Molecular atmospheric transmission from top of atmosphere to height z (Eqn 3.13).

$T_o(z)$ – Ozone transmission: top of atmosphere to height z (See section 3.3.3).

S_{ret} – Receiver return sensitivity (photons/J) – ATL03 parameter

/ancillary_data/atlas_engineering/receiver/rx_return_sensitivity Note that each beam has its own sensitivity. This parameter contains the min, max, mean and standard deviation of the return sensitivity. We will use the mean value.

N_a – Number of shots summed (nominally 400)

$R(z)$ – aerosol scattering ratio

Note that Equation 3.15 will have to be computed for each of the 3 strong laser beams if the emitted laser energy is much different for each beam. Equation 3.15 is used to compute a profile of molecular scattering contribution from 60 km to 0 km ($P_m(z)$). From that profile, the molecular

scattering contribution (folded from above) to the measured ATLAS photon profile is computed as:

$$P'_m(z) = P_m(z+15) + P_m(z+30) + P_m(z+45) \quad (3.16)$$

For z between -1 and 20 km. Note that we must compute this quantity up to 20 km since the raw profile is being captured 13.5 km above the value of the DEM at the satellite location, and there will be times (which occur over elevated terrain) when we need the values between 13.5 km and 20 km. We do not need to go higher than 20 km since, that portion of the profile (if it exists) will be eliminated as part of the vertical alignment process (see section 3.3.6). Note also the height in third term in 3.16 will go above 60 km for $z > 15$ km. This is above the top height of the input MET data (60 km). The values of $P_m(z)$ for $z > 60$ should be set to $P_m(60)$.

Then the corrected raw photon count profile is:

$$S'(z) = S(z) - P'_m(z) \quad (3.17)$$

Where $S(z)$ is the raw photon count profile measured by ATLAS. Note that this process leaves the molecular scattering of the original profile ($S(z)$) intact. It only removes the molecular scattering folded down from above. Also note that the quantity subtracted from the right side of Equation 3.17 ($P'_m(z)$) will be a fractional value. The raw photon count ($S(z)$) is an integer value and at times will be zero. Thus, the result ($S'(z)$) will at times be negative.

The NRB corrected for the molecular folding can now be computed as:

$$NRB'(z) = (S'(z) - p_b - p_d)r^2/E = C\beta(z)T^2(z) \quad (3.18)$$

Since we may not know all the instrument parameters accurately, or they may drift somewhat with time, a scale factor (α) is used in equation 3.15. If we knew all instrument parameters perfectly, the value of α would be 1. In practice it will not be unity. One way to find the value of α is to compute the average of the top 1 km of the raw (photon count) profile over the polar region (in darkness) and compare it with the average of that computed from equation 3.15 for the same vertical range. Over the poles, with no cloud or aerosol above 12.5 km, the average of the raw photon count between 12.5 and 13.5 km from the profile should match that computed from equation 3.15. Adjustment to α can then be made so that the ratio of the result of equation 3.15 to the 1 km profile top average is unity. The main problem with this method for determining α is knowing for sure that there is no cloud or aerosol scattering above 12.5 km. $R(z)$, the aerosol scattering ratio can be obtained from either GMAO model output or based on a climatological value computed from CALIPSO data as a function of latitude, height and season. $R(z) = 1.0 + \beta_a(z)/\beta_m(z)$ where $\beta_a(z)$ is the aerosol scattering coefficient at height z . Initially we will use $R(z) = \text{constant} = 1.02$.

In order to compute Equation 3.16, the molecular backscatter profile must be computed up to 60 km. Above 60 km (or the top height of the MET data) the molecular backscatter can be set to the

value at 60 km (or the top height of the MET data). Here we define the folded molecular backscatter profile as:

$$\beta'(z) = \beta(z) + \beta(z+15) + \beta(z+30) + \beta(z+45) \quad (3.19)$$

The folded molecular backscatter profile from $z=-1$ km to $z=20$ km (700 30 m bins) will be on the ATL04 and ATL09 products (parameter *Mol_Backs_folded*). For z below ground level, $\beta'(z)$ can be set to the lowest (in height) valid value.

3.3.2.1 Error Analysis of Molecular Contribution

The uncertainty related to the calculation of molecular contribution to the signal (equation 3.15) can be estimated as:

$$\left(\frac{\Delta P_m}{P_m}\right) = \sqrt{\left(\frac{\Delta E}{E}\right)^2 + \left(\frac{\Delta(\beta_m T_m)}{\beta_m T_m}\right)^2 + \left(\frac{\Delta T_o}{T_o}\right)^2 + \left(\frac{\Delta Q_e}{Q_e}\right)^2 + \left(\frac{\Delta T_r}{T_r}\right)^2 + \left(\frac{\Delta R}{R}\right)^2 + \left(\frac{\Delta \alpha}{\alpha}\right)^2} \quad 3.20$$

Values of the above equation will be included in a future version of this ATBD.

3.3.3 Ozone Transmission Computation

The first step in computing the ozone transmission is to calculate an atmospheric density profile following the methods presented in section 3.3.1 above. Equation 3.6 computes the molecular number density ($N(z)$). Replacing the Boltzmann constant (k) with the ideal gas constant for dry air (R) in equation 3.6, yields the atmospheric density:

$$\rho(z) = \frac{P(z)}{RT(z)} = \frac{k}{R} N(z) \quad (3.21)$$

The ozone transmission, $T_o^2(z_c)$, is calculated using ozone mass mixing ratios obtained from the GMAO meteorological data set which contains ozone mass mixing ratios. This, like other MET parameters are given at specific pressure levels. They are first linearly interpolated to 30 m bins as temperature was in section 3.3.1 forming a profile from 60 km to the ground ($r_o(z)$). The ozone mass mixing ratio profile is then converted to column density per kilometer (atm-cm/km), $\epsilon_o(z)$, using the following equation

$$\epsilon_o(z) = \frac{r_o(z) \rho(z)}{2.14148 \times 10^{-5}} \quad (3.22)$$

where z is the altitude in km, and $\rho(z)$ is the atmospheric density at z and calculated from equation 3.21.

The next step is to calculate the ozone transmission term. $T_o^2(\lambda)$ is calculated using the following equation:

$$T_o^2(\lambda, z) = \exp\left[-2c_o(\lambda) \int_H^z \varepsilon_o(z') dz'\right] \quad (3.23)$$

where $c_o(\lambda)$ is the Chappius ozone absorption coefficient in cm^{-1} . The ozone absorption coefficient is obtained at the correct wavelength from a table compiled in Iqbal [1984] using data from Vigroux [1953]. $c_o(\lambda)$ is 0.065 cm^{-1} at 532 nm. H is nominally 60 km.

The ozone transmission to the top of the calibration zone (z_c) is then corrected for the off-nadir angle of the laser beam (θ - different for each of the 3 strong beams):

$$T_o^2(\lambda, z_c) = T_o^2(\lambda, z_c)^{\sec(\theta)} \quad (3.24)$$

Where λ is 532 nm. The calculated ozone transmission (down to the top of the calibration zone – nominally 13.5 km) will be on the ATL04 product (parameter *Cal_Ozone_Trans*). A nominal value for the ozone transmission at the 13 km altitude is about 0.97.

3.3.4 Background Computation

As discussed in the introduction, the background computation would normally be done using the data below the ground. However, both molecular and cloud scattering may exist within this region from pulse aliasing (due to the 10 KHz laser rep rate). Hence, an alternative approach for background computation must be devised. There really is no way to do it accurately, unless one could remove all atmospheric scattering from the profile (which is essentially impossible). Here we present **three** separate methods for background computation using the atmospheric profiles. At present we feel both should be implemented and the results stored on the product. The background to be used in the computation of NRB (Equation 3.2) will be selected by the ancillary input parameter “Backg_Select” (see Table 3.5) which will also be stored on both the ATL04 and ATL09 products.

3.3.4.1 Background Method 1

For version V8.0 of the ATBD we introduce a modified algorithm for background computation. This new algorithm replaces method 1 discussed in prior ATBD versions, and will still be known as “method 1”. Examination of post launch data revealed that this method (method 1), while giving acceptable results during the day, was not providing good results during nighttime. Experimentation with the post-launch data has shown that for nighttime, it is best to set the background to a constant value and not compute it from the data. Thus, the new algorithm will require as input the solar elevation angle. Three regimes are considered in the algorithm: 1) night time when the solar elevation angle is less than -7.0 degrees, 2) daytime when the solar elevation angle is > -1.0 degrees and 3) twilight when the solar elevation angle is between these values.

During daytime, the background computation of the new algorithm is similar to that of old method 1.

For this method, 14 ASAS adjustable constants are used (default values are shown).

```
backg_night_scale_factor = 1.00600  
backg_day_scale_factor1 = 0.0100  
backg_day_scale_factor2 = 0.990  
backg_min_solar_elev = -7.0  
backg_max_solar_elev = -1.0  
backg_night_background = 0.0600  
backg_twilight_scale_factor = 0.6  
backg_nseg = 6  
backg_day_exp_factor = 8.5  
backg_response_time = 10.0e-9
```

The algorithm is based on 3 distinct solar elevation regimes: 1) night, 2) day and 3) twilight. The algorithm has been coded in IDL and tested and is supplied in Appendix C. It is suggested that the coders use this code as a guide.

1) Night:

When the solar elevation angle is less than or equal to *backg_min_solar_elev*, then set the background output from the algorithm (product parameter *backg_method1*) to *backg_night_background * backg_night_scale_factor*.

When in nighttime, an additional test is performed to determine if the orbit track is within an area that is affected by the South Atlantic Anomaly (SAA). This is a relatively large region extending from the South Atlantic Ocean about halfway between Africa and South America to western South America. This region is defined by a geographical box whose boundaries are:

```
saa_latmin = -40.0 ; Note that these should be made adjustable constants  
saa_latmax = -0.0  
saa_lonmin = -73.0  
saa_lonmax = -20.0  
saa_scale_fac = 0.60
```

If the current location of the data is within this area, then we compute the background from the data itself as shown in the IDL code snippet below:

```
b1 = total(profile(0:16,rec), 1) / 17.0 * saa_scale_fac ;profile is the raw atmosphere profile  
(photons)  
b2 = total(profile(nbt-17:nbt-1,rec), 1) / 17.0 * saa_scale_fac ; nbt is the number of bins in the  
profile  
background(rec) = b1  
if (b2 lt b1) then background(rec) = b2
```

2) Day:

The day portion of the algorithm can be summarized by the following steps:

- a) When the solar elevation angle is greater than *backg_max_solar_elev*, we use the whole 467 bin profile minus the bottom 33 bins (434 bins) and divide it into *backg_nseg* segments and compute the average of each segment. Let *seg_min* be the minimum segment average. Compute the factor *dtfac* as:

$$dtfac = 1.0 / (1.000 - backg_response_time * seg_min / 0.20e-6 / nshot_sum)$$

Where *backg_response_time* is the adjustable constant defined above, and *nshot_sum* is the number of ATLAS shots summed in one raw atmospheric profile (400) and 0.20e-6 is the atmospheric bin width in seconds.

- b) Compute: $fac = dtfac^{backg_day_exp_factor}$
- c) Set the *background* to *seg_min*. If the background is less than *backg_night_background*, then set *background* = *backg_night_background*.
- d) Compute: $badd = background * backg_day_scale_factor1$
- e) Set $background = background + (badd / fac)$
- ~~f) If the solar_elevation lt 0.0 then set $background = background * backg_day_scale_factor2$~~
- g) Finally, set the ATL04 product variable *Backg_Method1* to *background*.

3) Twilight:

If the solar elevation angle is greater than *backg_min_solar_elev* AND less than or equal to *backg_max_solar_elev*, it is considered twilight. The nominal range of solar elevation angles defined as twilight are relatively small (-7.0 to -1.0). In this solar angle range, we compute the background from the average of the top 500 m of the profile and apply a scaling factor that depends on the cosine of the solar elevation angle. The following IDL code segment shows how the background is to be computed during twilight:

```
fac = cos(8.0*solar_elev(ptr)*rpd) * 1.05
b1 = total(profile(0:16,rec), 1) / 17.0 * fac
b2 = total(profile(nbt-17:nbt-1,rec), 1) / 17.0 * fac
sval = b1
if (b2 lt b1) then sval = b2
dtfac = 1.0 / (1.000 - backg_response_time * sval / 0.20e-6 / nshot_sum)
background(rec) = sval
```



```

if (background(rec) lt backg_night_background) then background(rec) =
backg_night_background
badd = background(rec) * backg_day_scale_factor1
fac = dtfac^backg_day_exp_factor
background(rec) = background(rec) + (badd / fac)
if (background(rec) lt backg_night_background) then background(rec) =
backg_night_background

```

~~*background = total(profile(0:32,rec), 1) / 33.0 * backg_twilight_scale_factor*~~
~~if *background* is less than *backg_night_background* then set *background* =~~
~~*backg_night_background*, where *profile* is the raw atmospheric profile.~~

Note: because of this change to background method 1, we will no longer need the following ATL04 parameters: *Backg_Max_Bin1*, *Backg_Max_Bin2*, *Backg_Signal_Max1*, *Backg_Signal_Max2* and *Backg_Std_Dev1* on the ATL04 product. And the adjustable input constant *Backg_F1* can be removed and replaced by the 14 adjustable constants defined above.

3.3.4.2 Background Method 2

The following is a second method for background computation. Note that this method also uses the modified raw profile ($S'(z)$) as defined by equation 3.18.

- 1) Compute the mean and standard deviation of the $S'(z)$ profile. Call this S_{bar} and S_{std2} , respectively.
- 2) Set all bins that are greater than $S_{\text{bar}} + S_{\text{std2}} * \text{Back_F2}$ to -1. Where *Back_F2* is an adjustable, read-in constant (see Table 3.5).
- 3) Set all bins below 3 km to -1 in the profile. To eliminate overlap region with highest probability of cloud contributions
- 4) Compute (using equation 3.15) and subtract from the profile the molecular photon contribution for each bin.
- 5) Compute the average of the bins in the profile that are greater than -1.0. This is the background (p_{b2}).
- 6) If p_{b2} is negative, set to 0.0.

The values of S_{std2} , S_{bar} and p_{b2} should be stored on the ATL04 product. These values correspond to the product variables (in Table 3.1) *Backg_Std_Dev2*, *Backg_Mean2* and *Backg_Method2*.

3.3.4.3 Background Method 3

A third alternative for background (ATL04 parameter *Backg_Method3*) is the use of the onboard 50 shot background rate parameter originating from ATL03. This parameter (*bckgrd_rate* – see the ATL03 ATBD for details) has units of photons per second and must be converted to units of per atmospheric bin which is 30 m or 1/5 of a microsecond (0.20×10^{-6} s). Since there are 8 of these 50 shot values per the 400 shots that comprise each atmospheric profile, we average the 8 values and multiply by 400 to obtain a value consistent with the 400 shots summed to produce the atmospheric profile.

$$Backg_Method3 = bckgrd_rate * 0.20 \times 10^{-6} * 400.0$$

Where *bckgrd_rate* is actually the average of the 8 ATL03 *bckgrd_rate* values. Also stored on ATL04 should be the ATL03 parameters *Bckgrd_counts_reduced*, *Bckgrd_int_height*, *Bckgrd_rate*, and *Bckgrd_int_height_reduced*.

The parameter *Backg_Select* will be read in from an ancillary file and will control which background is used in equation 3.2 to compute the NRB. *Backg_Select*=1 will use background method 1. *Backg_Select*=2 will use background method 2 and *Backg_Select*=3 will use background method 3. *Backg_Select* will be on both the ATL04 and ATL09 products.

The background algorithm (method 2) has been coded and tested using simulated ATLAS data (produced from GLAS data). Folding of clouds and molecular scattering has been included in these simulations. The results of the background computation are shown in Figure 3.2. The algorithm output is plotted versus the actual background used in the model to produce the simulated data. It is apparent that the algorithm cannot reproduce exactly the model background, but it does a reasonable job. This can be seen in Figure 3.3. The top panel represents the ATLAS simulated data with the actual (model) background subtracted. The bottom panel is again the ATLAS simulated data, but the background subtracted was computed using the method 1 algorithm. Comparison of the two images suggests the background algorithm introduces some additional noise into the data, but that its magnitude is not large.

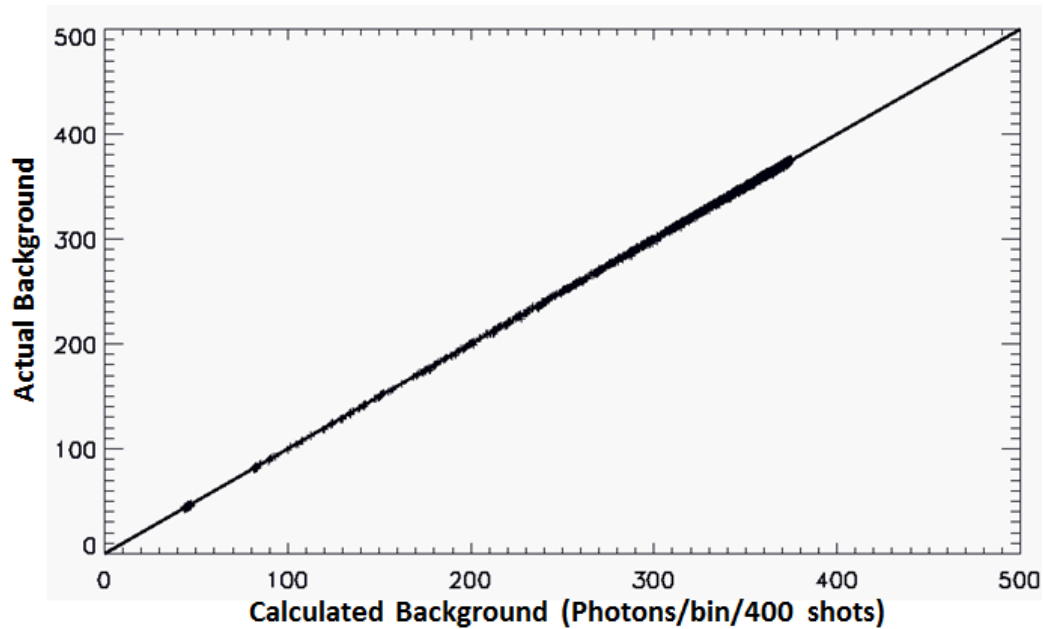


Figure 3.2 The actual (model) background versus the calculated background using the method 1 algorithm.

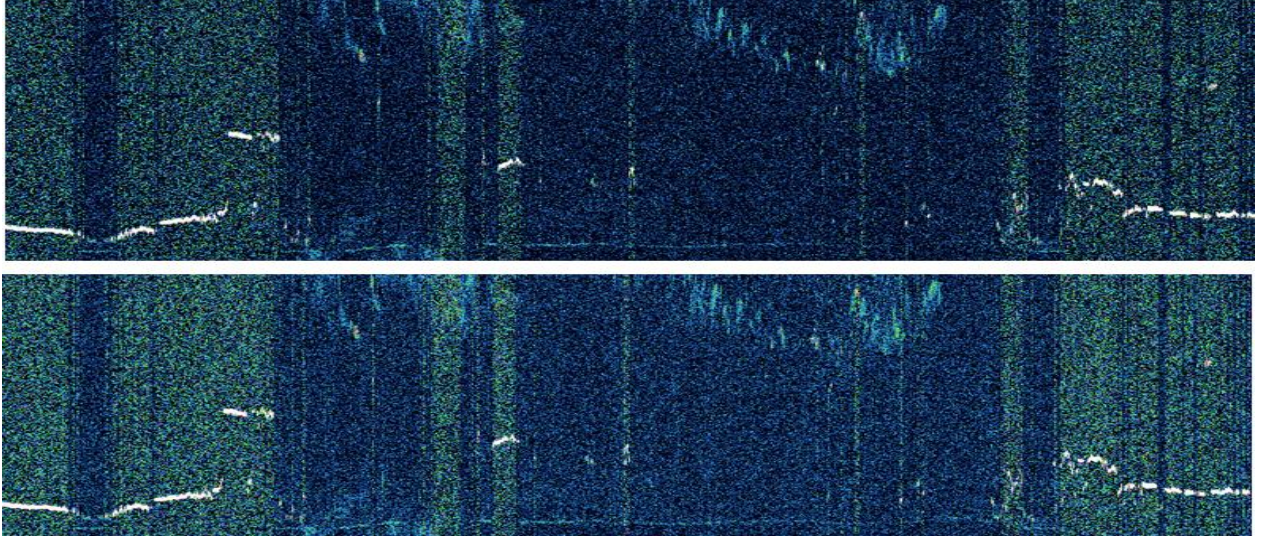


Figure 3.3. Top panel: The simulated ATLAS data with the actual (modeled) background subtracted. Bottom panel: Same data but with background subtracted computed using algorithm method 1.

Table 3.3. Background related parameters on the ATL04 product

Symbol used in background discussion	Suggested ATL04 Product Variable Name
background	Backg_Method1
S_{std2}	Backg_Std_Dev2
S_{bar}	Backg_Mean2
p_{b2}	Backg_Method2

Table 3.4. Ancillary constants used in this section. Note that some of the ATLAS instrument parameter values may change slightly by launch.

Constant	Value
Speed of light (c)	$2.9979 \times 10^8 \text{ ms}^{-1}$
Planck constant (h)	$6.6256 \times 10^{-34} \text{ Js}$
Bin size (Δz)	30 m
Effective collection area of telescope (A_t)	0.502655 m^2
Detector quantum efficiency (Q_e)	0.15
Transmission of receiver optics (T_s)	0.34
Number of laser shots summed (N_a)	400
King Factor	1.0401

Table 3.5 Adjustable parameters used in this section.

Constant/Adjustable Parameter	Use/Meaning	Nominal Value
backg_night_scale_factor	Scaling factor for nighttime background	1.00600
backg_day_scale_factor1	scaling factor for daytime background	0.0100
backg_day_scale_factor2	scaling factor for daytime background	0.990
backg_min_solar_elev	Maximum Solar elevation angle for data to be classified as nighttime	-7.0
backg_max_solar_elev	Minimum solar elevation angle for data to be classified as daytime	-1.0
backg_night_background	Value to be used for background when data is classified as nighttime	0.0600
backg_twilight_scale_factor	Scaling factor to be applied in calculation of background in twilight conditions	0.60
backg_nseg	Number of segments to break the atm histogram into for computation of daytime background	6
backg_day_exp_factor	Factor applied to daytime background	8.5
backg_response_time	Used for appx dead time correction of background	10.0e-9
saa_latmin	Latitude minimum of box that encompasses the area affected by the South Atlantic Anomaly (SAA)	-40.0
saa_latmax	Latitude maximum of box that encompasses the area affected by the South Atlantic Anomaly (SAA)	0.0
saa_lonmin	Longitude minimum of box that encompasses the area affected by the South Atlantic Anomaly (SAA)	-76.0
saa_lonmax	Longitude maximum of box that encompasses the area affected by the South Atlantic Anomaly (SAA)	-20.0
Backg_F2	Scaling factor in Method 2 background computation	2.5
α	Scaling factor for photons folded into the profile due to molecular scattering (Eqn 3.15)	See Table 3.2a
Backg_Select	Background method to use in computation of NRB	1
Dtime_Select	Defines which dead time correction factor to use (dtime_fac1 or dtime_fac2)	1

3.3.4.4 Theoretical Background

The solar background measured by ATLAS in clear sky is a function of the surface reflectivity, solar angle, atmospheric transmission and instrument parameters like the telescope field of view, filter width and telescope size. We want to compute the expected or theoretical background as it might provide useful for cloud detection during the day. This new parameter (for version 7.5) will be called *backg_theoret* and must be calculated by and stored on ATL09, since its computation relies on an estimate of the surface reflectivity. Surface reflectivity (*aclr_true*) is

computed and stored on ATL09 – see section 4.6.2.2. ATL04 does not have access to *aclr_true*. The equation below gives the ATLAS theoretical background for a clear atmosphere and assuming a Lambertian surface reflectance (following Ismail and Browell, 1989).

$$B_t = \frac{B_{ret} \cos(S_a) S_i R_s T_{fov}^2 T_a F_w A_{tran}^{(1+Sec(S_a))/2}}{4} \quad (3.25)$$

Where:

B_t = ATLAS theoretical background in photons/second

S_a = Solar zenith angle (degrees)

S_i = Solar irradiance at top of atmosphere at 532 nm (1.84d0 W/(m²*nm))

R_s = Reflectivity of the surface (use ATL09 parameter *aclr_true*)

T_{fov} = Telescope field of view (radians)

T_a = Telescope area (m²)

F_w = Interference filter width (nm)

A_{tran} = Two-way atmospheric transmission (0.81)

B_{ret} = Overall receiver sensitivity to background light (photons/J) (parameter /ancillary_data/atlas_engineering/receiver/rx_bckgrd_sensitivity from ATL03)

The parameter *backg_theoret* is then set to $B_t * 0.2e-6 * 400.0$ to get it in terms of photons per 30 m bin per 400 shots.

Note that in equation 3.25 we are using the ATL09 parameter *aclr_true* for the surface reflectivity term (R_s). In the computation of *aclr_true*, we have already taken into account the two-way atmospheric transmission at 532 nm. We thus must multiply the result of equation 3.25 by 1.0/T, where T is the two-way molecular transmission from the top of the atmosphere to the surface. In the determination of the height of the surface, the ATL04 parameter *surface_height* can be used when a surface signal has been detected. If a surface has not been detected (i.e parameter *surface_height* is invalid), then the height of the surface as indicated by the DEM should be used (parameter *dem_h*).

3.3.5 Surface Signal

The surface signal is intended to be all of the photons reflected from the surface. There are two ways to obtain the surface signal. The first is by searching the atmospheric profile as described below and the second is by using the surface signal photons that are detected via the ATL03 algorithm (see section 2.1 for a discussion of these ATL03 parameters). Here we outline the procedure to locate the surface signal from the atmospheric profiles.

The data used for the search should be the raw atmospheric profile (**raw_profile**) without the background subtracted. The algorithm will search in a narrow window about 300 m wide for the ground signal return using the raw photon count data. The search will start roughly 5 bins below the approximate surface bin (the bin that contains the surface return), which is calculated from

the surface height from the DEM and the top height of the profile. The search then continues upward for about 10 bins, looking for a bin that exceeds a surface signal threshold. Once that bin is found, the bin immediately above that bin is examined to determine if part of the surface return may be in that bin as well. The splitting of the surface return between two or more bins happens most often over rough or highly sloping terrain, but it can also happen over a flat surface if the return occurs at or very near a bin boundary. The search for high signal level can proceed up to 4 bins above the bin that contains the maximum signal. The algorithm is described below:

Inputs required: The raw atmospheric profile (**raw_profile**: the raw photon counts before background subtraction – it is OK to have molecular folding already subtracted from the profile, but it is preferred if it is not), the height above the ellipsoid of the top of the atmospheric profile (**profile_top_ht**), the solar elevation, the DEM (**dem_h**) and the adjustable parameters in Table 3.3.5.

Outputs: The product parameters *Surface_Bin*, *Surface_Height*, *Surface_Sig*, *Surface_Thresh* and two new parameters for ASAS 5.4 – *Surface_Width* and *Surface_Conf*. **NOTE: ALL of these parameters are to be put on both ATL04 and ATL09**

The algorithm has been implemented and tested in IDL. The code is listed in Appendix D. Note that this algorithm is applied to each of the 3 atmospheric profiles.

Method:

1. Set **fgrnd** = 0, *Surface_Signal* = 0.0, *Surface_Bin* = invalid, *Surface_Height* = invalid, *Surface_Thresh* = 0.0, *Surface_Conf* = 0 and *Surface_Width* = 0
2. Break the profile into 5 equal length segments and compute the mean and standard deviation of each segment.
3. Find the segment with the minimum average and set the variable **min_seg_avg** to the segment average and the variable **min_seg_std** to that segment's standard deviation.
4. Define a signal level, call it **slev**, and set **slev** = **min_seg_avg** + **min_seg_std** * 3.0.
5. Define the variable **backg** as: **backg** = **min_seg_avg**
6. Define a bin called **ground_bin** = fix((**profile_top_ht** - **dem_h**) / 30.0)
7. Interrogate bins 0 to **ground_bin**-10 of the raw profile and set bins that are greater than **slev** to **backg**.
8. Set the variable **sdfac** = **grnd_thresh1_sf**. If the solar elevation is greater than 0.0, then set **sdfac** = **grnd_thresh2_sf**
9. Subtract **backg** from all bins of **raw_profile**. Call the resulting profile **backg_subtracted_profile**.
10. Compute the standard deviation of **backg_subtracted_profile** from bin 0 to **ground_bin**-10. Call this **stdev**
11. Set **thr1** = **sdfac** * **stdev**. If **thr1** < 4.0, then set **thr1** = 4.0
12. Set *Surface_Thresh* = **thr1**
13. Set **bin1** = **ground_bin** - **grnd_search_width**/2 - 1 and **bin2** = **ground_bin** + **grnd_search_width**/2 and **fgrnd** = 0

14. Interrogate the bins **bin1** to **bin2** of **backg_subtracted_profile** going from **bin2** to **bin1** (moving upwards in the profile) and find the first bin that exceeds **thr1**. If found, call this **max_bin0** and set **max_sig0** = **backg_subtracted_profile(max_bin0)**. Also set **Surface_bin** = **max_bin0** and **fgrnd** = 1 and set **limit** to **max_bin0** - 3
15. If **fgrnd** equal 0, then go to step 29
16. Set **max_sig1**, **max_sig2**, **max_sig3** and **max_sig4** to 0.0; set **Wide_Ground** = 0
17. Using **backg_subtracted_profile**, find the maximum signal of the bins between **max_bin0** and **limit**. Call the corresponding bin **max_bin**
18. Set **Surface_Conf** to float (**backg_subtracted_profile(max_bin)**) / **thr1**. If **Surface_Conf** > 100.0, then set **Surface_Conf** = 100.0
19. If **max_bin** == **max_bin0** go to step 21
20. Set **max_sig1** to the summation of the signal in all bins between **max_bin0**-1 and **max_bin** in **backg_subtracted_profile** and set **Wide_Ground** = 1
21. Set **thr2** = **backg_subtracted_profile(max_bin)** * 0.50
22. If **thr2** < 30.0 * **thr1**, then **thr2** = 30.0 * **thr1**
23. If **backg_subtracted_profile(max_bin-1)** > **thr2** AND **max_bin** - 1 >= **limit**, then do steps 24 to 26. Otherwise go to step 27
24. set **max_sig2** = **backg_subtracted_profile(max_bin-1)**, **Surface_Bin** = **max_bin** - 1 and increment **Wide_Ground**
25. If **backg_subtracted_profile(max_bin-2)** > **thr2** AND **max_bin** - 2 >= **limit** then set **max_sig3** = **backg_subtracted_profile(max_bin-2)**, increment **Wide_Ground** and set **Surface_Bin** = **max_bin** - 2. Otherwise go to step 27
26. If **backg_subtracted_profile(max_bin-3)** > **thr2** AND **max_bin** - 3 >= **limit** then set **max_sig4** = **backg_subtracted_profile(max_bin-3)**, increment **Wide_Ground** and set **Surface_Bin** = **max_bin** - 3
27. Set **Surface_Height** = **profile_top_ht** - **Surface_Bin** * 30.0, **Surface_Signal** = **max_sig0** + **max_sig1** + **max_sig2** + **max_sig3** + **max_sig4** and **Surface_Width** = **Wide_Ground** + 1
28. If **Surface_Signal** < 0.0, then **Surface_Signal** = 0.0
29. Set **Surface_Signal** = **Surface_Signal** * **Dtime_fac1**
30. Done examining this profile. Go to next profile.

Table 3.3.5. Adjustable parameters for ground search algorithm

Parameter	Default Value
grnd_thresh1_sf	5.0
grnd_thresh2_sf	5.0
grnd_search_width	10
Dtime_max	10.0
Dtime_select	1

The computation of *dtime_fac1* and *dtime_fac2* was presented in section 2.1. The dead time correction factor is a number greater than or equal to one but is not to exceed *dtime_max*. *Dtime_max* is a unitless float with nominal value = 10.0 and denotes the maximum dead time

correction value to be applied to the surface signal. The surface signal is corrected for dead time by multiplying it by the dead time factor after making sure it is less than *dtime_max*.

3.3.6 Vertical Height Adjustment

The raw atmosphere profiles will be captured based on the value of the DEM at the sub-satellite point. They will extend from about 500 m below the DEM value to roughly 13.5 km above (14 km total). The height of the top bin can shift from one 25 Hz profile to the next as it follows the DEM (but more likely at a slower rate). Before horizontal averages of the data can be made (needed for section 3.3.7 - calibration), they must be put in a constant height reference frame (vertical height adjustment). The profiles will be placed into a constant reference frame with respect to the WGS84 ellipsoid (or mean sea level). The resulting 25 Hz profiles (and the subsequent NRB that will be stored on the ATL04 product – parameter *NRB_Prof* in Table 3.1) will cover the vertical range from -1 km to 20 km altitude and consist of 700 bins. Thus, over the oceans and low terrain, the data above about 13.5 km will be set to an invalid value, as will the data below about -500 m. When we are over elevated terrain, some of the bins above 13.5 km will then be populated. For instance over the high plateau of Antarctica, surface height (with respect to the WGS84 ellipsoid) can approach 4 km. In this case, the -1 to 20 km profile would have invalid values between -1 and about 3.5 km, data between 3.5 km and 17.5 km, and invalids between 17.5 and 20 km. As part of the vertical alignment algorithm, the bin number (starting from 1 at the top – call this parameter *NRB_Top_Bin*) of the first valid data bin (within the 20 to -1 km data frame) will be stored on the product for each profile. The last valid data bin is then *NRB_Top_Bin* + 466. It is acknowledged that when DEM values are greater than 6 km, data will be cut off at the top. Horizontal averages using the same bin number will then be at the same altitude. However, care must be taken to avoid invalid data bins when summing or averaging.

Since the pointing angles of the 3 strong beams will be on the order of 0.3 – 0.5 degrees with respect to nadir, vertical shifting of the bins will be negligible. However, at times ICESat-2 will off point to no more than 5 degrees for (altimetry) calibration scans. Even in these cases, the change in vertical size of the bin is only 12 cm which amounts to a 44 m change in the overall vertical extent of the profile. The pointing angle will be considered in the algorithm to place the data in the constant height reference frame. The best way to describe such an algorithm is by the (IDL) code segment below:

```
Nbins_atlas = 467      ; Number of bins in the raw (level 0) ATLAS data.
Nbins_frame = 700      ; Number of bins in constant reference frame (20 to -1 km wrt WGS84
ellipsoid)
Frame = fltarr(Nbins_frame)
Frame(*) = 9999.        ; Set all bins in Frame to invalid value
ATLAS_profile = fltarr(Nbins_atlas)
Frame_top = 20.0
Bin_size = 0.030
ATLAS_altitude = 495.0  ; Height of satellite in km wrt WGS84 ellipsoid (supplied to this
algorithm)
```


Range_to_data_start ; The range from satellite to the top of the atm profile (ATL02?)
Height_top_bin = ATLAS_altitude - Range_to_data_start ; This can range from 13.5 to over
; 20 km depending on DEM value

Pointing_angle = y ; This is the pointing angle off nadir of spacecraft from ATL02?
Frame_ht = Height_top_bin
Bin_size_angle = Bin_size * cos(Pointing_angle)

```
i1 = 0
If (Frame_ht gt Frame_top) then begin
    i1 = fix((Frame_ht - Frame_top) / Bin_size_angle)
    Frame_ht = Frame_top
endif
```

```
For i=i1,Nbins_ATLAS-1 do begin
    Frame_bin = fix((Frame_top - Frame_ht) / Bin_size)
    If (Frame_bin ge Nbins_frame) then Frame_bin = Nbins_frame - 1
    Frame(Frame_bin) = ATLAS_profile(i)
    Frame_ht = Frame_ht - Bin_size_angle
Endfor
```

Note that it is possible for this process to skip a bin in the frame leaving the bin undefined. After the above is executed it is recommended to add the following code that interrogates all bins in the vertically aligned frame:

```
do i = va_first_bin, va_last_bin
    if (atm_va_bins(i) == INVALID_R4B) then
        if (i-1 >= va_first_bin .and. atm_va_bins(i-1) /= INVALID_R4B) then
            atm_va_bins(i) = atm_va_bins(i-1)
        else if (i+1 <= va_last_bin .and. atm_va_bins(i+1) /= INVALID_R4B) then
            atm_va_bins(i) = atm_va_bins(i+1)
        endif
    endif
enddo
```

3.3.7 Calculation of the Calibration Coefficient

3.3.7.1 Theory

The following discussion is not required for the coded algorithm. It is merely meant as an introduction to the meaning and use of the lidar calibration coefficient. The lidar calibration coefficient relates the power received by the detector to a physical quantity – the volume backscatter cross section. The theoretical or instrument lidar calibration coefficient (C_l) is mainly a function of the transmission of the system optics (transmitting and receiving), detector

efficiency, geometric considerations, and the degree of alignment between the receiver field of view and the laser spot (boresite alignment). The following equation gives the theoretical derivation of the lidar calibration coefficient.

$$C_t = FA_t T_s Q_e \lambda \Delta z / (hc) \quad (3.26)$$

Where A_t is the telescope area, c the speed of light, h the Planck constant, T_s the receiver system transmission, Q_e the quantum efficiency (or responsivity) of the detector, Δz is the bin size, λ the laser wavelength and F is the boresite factor. If the system is perfectly boresited, $F=1$. As the receiver field of view becomes misaligned with the laser footprint (some of the laser energy is outside of the telescope FOV), F becomes less than 1. In practice, it is mainly thermally driven changes in boresite that cause changes in C_t . However, changes in detector responsivity, and system transmission are also common causes for variation in C_t . Because these factors can and do change with time in the typical lidar system, and these changes cannot easily be monitored, calibration of the lidar signal is usually performed continuously by comparing the measured signal to a reference target. The atmosphere itself can act as a reference target so long as it is devoid of particulate scatterers (or the magnitude of particulate loading is known) and the density of the atmosphere is sufficiently well known. For instance, in the lower and mid stratosphere, in the absence of large volcanic eruptions, the air is typically very clean. The scattering there consists only of molecular or Rayleigh scattering which depends on only the pressure and temperature of the atmosphere. We can very accurately calculate the molecular backscatter cross section (i.e. the target) in near real-time from NCEP (National Center for Environmental Prediction) or GMAO (Goddard Modeling and Assimilation Office) meteorological analyses or short range forecasts of temperature, pressure and moisture fields.

3.3.7.2 Calibration Algorithm using the Atmosphere

NOTE: In the discussion below, the algorithm adjustable parameters are in **blue font** and are assumed read in by the SIPS code from an ancillary file. These are summarized in Table 3.8 below.

V8.0 Note: After evaluating post launch data for the first time, it was apparent that the calibration algorithm as initially designed was not going to work well. This is mainly due to the fact that it computes the calibration constant during night only, and then interpolates it through the day portion of the orbit. It was found that the calibration constant changed considerably during daylight, making this approach untenable. This document, the first post-launch version of the ATBD, introduces 2 new calibration methods in addition to the original one (for a total of 3 calibration methods. The calibration method to use will be selected by a control parameter – *calib_select*. When *calib_select*=1, the current algorithm now called “Calibration Method 1” is invoked. When *calib_select*=2, new calibration method called “Calibration Method 2” is invoked. When *calib_select*=3, new calibration method called “Calibration Method 3” is invoked. The default value for *calib_select* will be 2.

Below we describe the 3 calibration methods in order.

Calibration Method 1 (the original algorithm)

GLAS successfully used the 26 to 30 km region of the stratosphere for calibration of its 532 nm channel. CALIPSO uses an even higher region for its calibration. Unfortunately, this portion of the atmosphere will not be available in ATLAS data. We will be forced to calibrate in the region 13.5-11 km. This severely limits where we can perform the calibration, since clouds typically occur up to 18 km height in the tropics and subtropics. However, for latitudes poleward of 60 – 65 degrees, clouds are generally confined below 10 – 11 km. The exception to this occurs in late winter when Polar Stratospheric Clouds often form in the stratosphere (most frequently over Antarctica). Thus, while this region offers the best hope for calibration, the data will have to be carefully screened to avoid the inclusion of cloud signal into the calibration procedure. This will be accomplished by setting a TBD threshold – call this [Cal_Cloud_Thresh](#) - on the value of the integrated signal between 11 and 13.5 km.

Due to the very weak molecular signal, we will most likely not be able to calibrate during daylight conditions. For each orbit we will use the data poleward of 60 degrees (exact latitude will be an adjustable algorithm parameter – call this [Cal_Latitude_Bound](#)) that have low background (determined by solar Zenith angle). The bins between 11 and 13.5 km will be interrogated and those that pass a threshold test to eliminate clouds (value will be adjustable – [Cal_Cloud_Thresh](#)) will be summed (horizontally and vertically). The result will be the calibration target. The length of time the data will be summed within the calibration zone will likely be at least 4-5 minutes. This summing length will be an adjustable algorithm parameter – call this [Cal_Integ_Time](#). While this sum is being computed, concurrently the average molecular backscatter for the same vertical window and horizontal distance will be computed from the ancillary meteorological (MET) data. The computation of the calibration coefficient then is:

$$C = \overline{NRB'(z_c)} / [\overline{(\beta_m(z_c))} T^2(z_c) R(z_c)] \quad (3.27)$$

Where $\overline{NRB'(z_c)}$ and $\overline{(\beta_m(z_c))}$

are the horizontally and vertically integrated normalized lidar signal (corrected for molecular folding – equation 3.19) and molecular backscatter through the 2.5 km thick calibration layer (11-13.5 km), respectively. These are known as product parameters NRB_C and $Molec_C$, respectively. Note that the molecular backscatter at z_c is simply from that height, since in the computation of NRB we have removed the folded molecular scattering.

The length of the horizontal average will vary according to the solar background conditions and the presence of cloud above 11 km altitude. In general we want to make this horizontal averaging length long enough to gather sufficient signal, but not too long so as to include varying atmospheric conditions along the horizontal averaging path. The exact averaging length is TBD

and will be an algorithm parameter ([Cal_Integ_Time](#)), initially set to 300 seconds (roughly 20 degrees of latitude). We will attempt to do this at least twice while the spacecraft is poleward of 60 degrees. In equation 3.27, the R factor is known as the aerosol scattering ratio. This also appears in equation 3.15. In a totally clean atmosphere devoid of cloud or aerosol is equal to 1.0. In practice it is always greater than 1.0 due to the presence of aerosol (except possibly in the mid stratosphere, 30-40 km altitude – an area we do not have access to). We will assume a value of R based on climatology. We hope to use data from the CALIPSO mission to help determine a nominal value. This will generally be between 1.01 and 1.20 and is an algorithm parameter – call this [Cal_Scat_Ratio](#). Closer to launch, we will analyze CALIPSO 532 nm data in the polar region to calculate a representative value to use which can vary seasonally.

The computation of $\overline{NRB'}(z_c)$ in equation 3.22 will proceed for each of the 3 strong beams by summing the NRB in the altitude range 11 to 13.5 km. Thus, there will be a calibration coefficient for each of the 3 strong beams. If this 2.5 km sum is greater than [Cal_Cloud_Thresh](#), then that sum will not be included in the calculation of $\overline{NRB'}(z_c)$. In addition, summing will only occur for those profiles that are obtained with the solar elevation angle less than [Cal_Solar_Angle_Limit](#) degrees (nominally 0.0 degrees). The sum will then be normalized by the total number of bins summed.

The molecular backscatter used in equation 3.27, will be computed once per second and computed for each 30 m bin in the calibration zone(s). One molecular backscatter value is then computed for the layer(s) by averaging the bins within the calibration zone(s). The final average molecular backscatter is then computed from the sum of these layer averages for the duration of the calibration time ([Cal_Integ_Time](#) - nominally 300 seconds). The latitude, longitude and time of the start of the calibration ([Cal_Start_Latitude](#), [Cal_Start_Longitude](#) and [Cal_Start_Time](#)) will be stored on the ATL04 product. The same average molecular backscatter value will be used for each of the 3 beams.

In equation 3.27, $T^2(z_c)$ represents the two-way path transmission from the top of the atmosphere to the calibration height and is composed of Rayleigh, particulate and ozone components as: $T^2(z_c) = T_m^2(z_c)T_p^2(z_c)T_o^2(z_c)$. $T_p^2(z_c)$, the particulate transmission term, will not be known exactly and must be estimated from climatology. The expected value of $T_p^2(z_c)$ will range between 0.90 and 1.0 and will be an algorithm adjustable parameter – call this [Cal_Atm_Trans](#). The molecular transmission term $T_m^2(z_c)$ is computed from the molecular backscatter as from equations 3.10, 3.11, 3.12 and 3.13 in section 3.3.1. T_m^2 has a value of about 0.81 at sea level but will be closer to 0.90 at the 13 km height. A representative value of the ozone transmission term would be about 0.97 at 532 nm. The procedure to calculate the ozone transmission, $T_o^2(z_c)$, is given in section 3.3.3.

NOTES: If for some reason, no calibration points could be calculated for method 1, then the calibration constant should be set to the default night value as defined in calibration method 2

below (parameter *calib_night_pce1*, 2 or 3). If the calculated calibration constant is less than the parameter *min_calib_night* (defined in Table 3.6), or if it is greater than parameter *max_calib_night* (defined in Table 3.6), then the calibration constant should be set to *calib_night_pce1*, 2, or 3 as defined in calibration method 2 and 3 below and Table 3.6.

Calibration Method 2

For this calibration method, we use a constant calibration factor (C) for daytime (new adjustable parameter *calib_day*), a constant calibration factor for night (new adjustable parameter *calib_night*) and a constant calibration factor for twilight (*calib_twilight*). **There will be a calibration constant for each of the 3 profiles as shown below.** When these values are used depends on the solar elevation angle. We introduce two more adjustable parameters that define the solar elevation angle limits: *calib_solar_elev_min* and *calib_solar_elev_max*. When the solar elevation angle is less than *calib_solar_elev_min*, then set C to *calib_night*. When the solar elevation angle is \geq *calib_solar_elev_max*, then set C to *calib_day*. When the solar elevation angle is between *calb_solar_elev_min* and *calib_solar_elev_max*, then set C to *calib_twilight*.

Nominal values of the calibration constants for the 3 beams are defined in Table 3.6 below.

The nominal values for the solar elevation limits are:

calib_solar_elev_min, = -6.0

calib_solar_elev_max = 0.0

Calibration Method 3

This calibration method computes the calibration constant using the average NRB between 13 and 11 km continuously for a complete orbit. Unlike calibration method 1, it does not limit the calibration calculation to high latitudes and only at night. This method computes the calibration constant continuously over the whole orbit, both day and night **for each profile**. In so doing it must screen out high values of NRB caused by clouds. It uses the NRB data from one complete orbit that has been vertically aligned as described in section x. This calibration method uses a number of adjustable parameters that are defined in Table 3.6. The parameters that are dimensioned by 3 correspond to the 3 atmospheric profiles as: (1) = *atlas_pce3*, (2) = *atlas_pce2* and (3) = *atlas_pce1*.

For ASAS version 5.4 (version 8.4 of this document), we introduce 2 new adjustable constants and a change to calibration method 3. The South Atlantic Anomaly (SAA) increases the background noise of the ATLAS and its location and effect on background is discussed in section 3.3.4.1. The increased noise while in SAA region is only a factor during nighttime. Thus, the nighttime limit on the NRB defined in Table 3.6 (*max_nrb_night*) does not apply while in the SAA. A new maximum NRB is now required while in the SAA region (*max_nrb_saa*) during

nighttime. The calibration method 3 code will then use this limit in step 3 below while in the SAA during nighttime only (see step 3 below).

Table 3.6. Adjustable parameters for calibration method 3.

Parameter	Type	Function	Default Value(s)
nrb_smooth	Scaler	Number of points to average the NRB data	50 (equivalent to 2 seconds for 25 Hz data)
default_nrb_day	Float(3)	Default value for daytime NRB - used to replace NRB values that are out of range	Atlas_pce3=9.5e14, Atlas_pce2=9.5e14, Atlas_pce1=9.5e14
default_nrb_night	Float(3)	Default value for nighttime NRB - used to replace NRB values that are out of range	Atlas_pce3=3.0e14, Atlas_pce2=1.5e14, Atlas_pce1=3.0e14
default_nrb_twilight	Float(3)	Default value for twilight NRB - used to replace NRB values that are out of range	Atlas_pce3=8.0e14, Atlas_pce2=8.0e14, Atlas_pce1=8.0e14
default_nrb_saa	Float(3)	Default value for NRB - used to replace NRB values that are out of range when in the South Atlantic Anomaly	Atlas_pce3=1.2e15, Atlas_pce2=1.2e15, Atlas_pce1=1.2e15
min_nrb_day	Float(3)	Minimum daytime NRB accepted for filtered NRB data	Atlas_pce3=-0.8e15, Atlas_pce2=-0.8e15, Atlas_pce1=-0.8e15
max_nrb_day	Float(3)	Maximum daytime NRB accepted for filtered NRB data	Atlas_pce3=2.0e15, Atlas_pce2=2.0e15, Atlas_pce1=2.0e15
min_nrb_night	Float(3)	Minimum nighttime NRB accepted for filtered NRB data	Atlas_pce3=1.0e14, Atlas_pce2=5.0e13, Atlas_pce1=1.0e14
max_nrb_night	Float(3)	Maximum nighttime NRB accepted for filtered NRB array	Atlas_pce3=5.0e14, Atlas_pce2=3.0e14, Atlas_pce1=5.0e14
max_nrb_saa	Float(3)	Maximum NRB accepted for filtered NRB data when in the South Atlantic Anomaly during nighttime	Atlas_pce3=2.2e15, Atlas_pce2=2.2e15, Atlas_pce1=2.2e15

min_nrb_twilight	Float(3)	Minimum twilight NRB accepted for filtered NRB data	Atlas_pce3=1.0e14, Atlas_pce2=1.0e14, Atlas_pce1=1.0e14
max_nrb_twilight	Float(3)	Maximum twilight NRB accepted for filtered NRB array	Atlas_pce1=5.0e14, Atlas_pce2=5.0e14, Atlas_pce3=5.0e14
nrb_average_period	Scaler	Number of seconds to average the smoothed and filtered NRB array before computing the calibration constant	90
calib_day NOTE this is the same constant as defined in calib method 2	Float(3)	Default daytime calibration value (same parameter as in method 2)	Atlas_pce3=1.5e21, Atlas_pce2=1.4e21, Atlas_pce1=1.7e21
calib_night NOTE this is the same constant as defined in calib method 2	Float(3)	Default nighttime calibration value (same parameter as in method 2)	Atlas_pce3=7.610e20, Atlas_pce2=4.5000e20, Atlas_pce1=7.92e20
calib_twilight NOTE this is the same constant as defined in calib method 2	Float(3)	Default twilight calibration value (same parameter as in method 2)	Atlas_pce3=1.815e21 Atlas_pce2=1.565e21 Atlas_pce1=1.815e21
min_calib_day	Float(3)	Minimum calculated calibration allowed for daytime	Atlas_pce3=1.0e21, Atlas_pce2=1.0e21, Atlas_pce1=1.0e21
max_calib_day	Float(3)	Maximum calculated calibration value allowed for daytime	Atlas_pce3=3.0e21, Atlas_pce2=3.0e21 Atlas_pce1=3.0e21
min_calib_night	Float(3)	Minimum calculated calibration value allowed for nighttime	Atlas_pce3=5.0e20, Atlas_pce2=1.0e20, Atlas_pce1=5.0e20

max_calib_night	Float(3)	Maximum calculated calibration value for nighttime	Atlas_pce3=1.8e21, Atlas_pce2=9.0e20, Atlas_pce1=1.8e21
min_calib_twilight	Float(3)	Minimum calculated calibration value allowed for twilight	Atlas_pce3=8.0e20, Atlas_pce2=6.0e20, Atlas_pce1=8.0e20
max_calib_twilight	Float(3)	Maximum calculated calibration value for twilight	Atlas_pce3=1.4e21, Atlas_pce2=1.2e21, Atlas_pce1=1.4e21
saa_cal_fac	Float	Factor to multiply the calibration constants by when within the SAA and the solar elevation angle is < <i>calib_solar_elev_max</i>	0.80 for all pce

For each of the 3 profiles do the following steps:

- 1) Compute the average NRB between 13 and 11 km for an entire granule from the 25 Hz NRB array computed from ATBD equations 3.2 and 3.17. Let “npoints” be the number of resulting points. This will nominally be the total number of NRB profiles in the granule.
- 2) Average the result of 1) by “*nrb_smooth*” points, creating another time series with the number of elements equal to $nrb_smooth_npoints = npoints / nrb_smooth$
- 3) Filter the result of 2) by eliminating those *nrb* values greater than *max_nrb_day* or less than *min_nrb_day* when the solar elevation angle is greater than or equal to *calib_solar_elev_max*, and those values greater than *max_nrb_night* or less than *min_nrb_night* when the solar elevation angle is less than *calib_solar_elev_min*. for solar elevations less than *calib_solar_elev_max* and greater than or equal to *calib_solar_elev_min* we filter the *nrb* values by eliminating those *nrb* values greater than *max_nrb_twilight* or less than *min_nrb_twilight*. When a value is eliminated, set the filtered array value to *default_nrb_day* or *default_nrb_night* or *default_nrb_twilight*, depending on the solar elevation angle. Because we are filling in eliminated values with default NRB values, this filtered time series will have the same length as the time series in 2). Call this array “*nrb_filtered*”. Note that while in the SAA region as defined in section 3.3.4.1 AND the solar elevation is less than *calib_solar_elev_max*, then do not check on the *nrb* values to see if they are greater than *max_nrb_saa* and replace them with *default_nrb_saa*.

- 4) Create the average attenuated molecular backscatter between 13 and 11 km at the same temporal resolution as the `nrb_filtered` array created in step 3). The molecular backscatter is computed at 1 Hz, so to do this for `nrb_smooth=50` (or 2 seconds), two attenuated molecular backscatter profiles would be averaged. Call the resulting array “`molback_11_13km`”.
- 5) Compute:

$$\text{interval} = \text{fix}(\text{nrb_smooth_npoints} / \text{orbit_period_seconds} * \text{nrb_average_period}).$$
Where `nrb_smooth_npoints` is defined in 2) above, `orbit_period_seconds` is the ICESat2 orbital period in seconds (appx 96*60), and `nrb_average_period` is defined in Table 3.6.
- 6) Smooth the `nrb_filtered` and `molback_11_13km` arrays by averaging “interval” number of points and compute the calibration array (`calib`) as in the code snippet below. The calibration constant is computed from ATBD equation 3.27:

$$C = \overline{NRB'(z_c)} / [\overline{(\beta_m(z_c))} T^2(z_c) R(z_c)] \quad 3.27$$

Where $\beta_m(z_c)$ is the attenuated molecular backscatter (`molback_11-13km`), $T^2(z_c)$ is the two way transmission from the top of the atmosphere to the height z_c , here taken to be 12 km. $T^2(z_c)$ in the above equation is composed of molecular transmission (which is already included in `molback_11_13km` array defined above), ozone transmission, $T^2_o(z_c)$, (defined in ATBD section 3.3.3) and the transmission loss due to particulates, $T^2_p(z_c)$.

$$T^2(z_c) = T^2_m(z_c) T^2_p(z_c) T^2_o(z_c)$$

$T^2_p(z_c)$, the particulate transmission term, will not be known exactly and must be estimated from climatology. The expected value of $T^2_p(z_c)$ will range between 0.90 and 1.0 and will be an algorithm adjustable parameter – call this `Cal_Atm_Trans`. In equation 3.27, the R factor is known as the aerosol scattering ratio. This also appears in equation 3.15. In a totally clean atmosphere devoid of cloud or aerosol is equal to 1.0. In practice it is always greater than 1.0 due to the presence of aerosol (except possibly in the mid stratosphere, 30-40 km altitude – an area we do not have access to). We will assume a value of R based on climatology. We hope to use data from the CALIPSO mission to help determine a nominal value. This will generally be between 1.01 and 1.10 and is an algorithm parameter – call this `Cal_Scat_Ratio`.

- 7) When computing the `calib` array, filter it by eliminating points greater than `max_calib_day` and points less than `min_calib_day` (for solar elevation angles \geq `calib_solar_elev_max`) and greater than `max_calib_night` and less than `min_calib_night` (for solar elevation angles $<$ `calib_solar_elev_min`). For solar elevation angles less than `calib_solar_elev_max` and \geq `calib_solar_elev_min` filter the `calib_array` values by eliminating points greater than `max_calib_twilight` and points less than `min_calib_twilight`. When a calibration point is eliminated, set the filtered point to `calib_day` or `calib_night` or `calib_twilight`, depending on the solar elevation angle. New for ATBD version 8.4: When in the SAA region as defined in section 3.3.4.1, AND the

solar elevation angle is less than `calib_solar_elev_max`, then DO NOT filter the calibration values. Instead they are to be multiplied by a scaling factor. This scaling factor is to be called `saa_cal_fac` and has a nominal value of 0.70.

NOTE: “`cal_scatter_ratio`” used in the below code is defined in table 3.7 of the ATBD and has a nominal value of 1.12.

Note: in the below code the variable “Trans” is equal to $T_p^2(z_c)T_o^2(z_c)$

Note also the below works on just one atmospheric profile.

Note also that `calib_twilight` in the below code is the constant defined in calibration method 2.

```
j = 0
lim = nrb_smooth_npoints
for i=0,lim-interval,interval do begin
  segavg_nrb = total(nrb_filtered(i:i+interval-1)) / float(interval)
  segavg_mol = total(molback_11_13km(i:i+interval-1)) /
float(interval)
  segavg_sa = total(avg_sa(i:i+interval-1)) / float(interval)
  if (segavg_nrb gt 0.0) then begin
    calib(j) = segavg_nrb / (segavg_mol * Trans * cal_scatter_ratio)
    if (segavg_sa ge calib_solar_elev_max) then $
      if (calib(j) lt min_calib_day or calib(j) gt max_calib_day) $
        then calib(j) = calib_day
    if (segavg_sa lt calib_solar_elev_min) then $
      if (calib(j) lt min_calib_night or calib(j) gt max_calib_night)
        then calib(j) = calib_night
    if (segavg_sa ge calib_solar_elev_min and segavg_sa lt calib_solar_elev_max
) then calib(j) = calib_twilight
    j++
  endif
endfor
```

- 8) Linearly interpolate the `calib` array to one second resolution.
- 9) For each bin of each 25 Hz NRB profile of a given second, compute the calibrated, attenuated backscatter as $NRB(i,j) / calib(k)$, where i is the i^{th} bin of the j^{th} NRB profile, and $k = \text{fix}(j/25)+1$.

3.3.7.3 Calibration Algorithm using Surface Reflectance

Note: this section is not to be implemented in the SIPS processing code. After launch, this technique will be investigated using actual ATLAS data. If it is deemed a viable approach to calibration, it may be incorporated into SIPS processing at a later date.

The calibration coefficient can in theory be calculated from the surface return if certain conditions are met. Both the surface reflectivity and the atmospheric transmission to the surface must be well known. The interior of East Antarctica provides one of the best places where both

of these conditions can be met. Over the ice sheet, the surface reflectivity is well known and fairly constant (to within about 5%). Additionally, the atmosphere is very dry and usually cloud and aerosol free which enables a good estimate of the atmospheric transmission. Another possibility is over the ocean where the surface reflectivity can be estimated from the surface wind speed (see section 4.7). However, over the ocean, it is much more difficult to characterize the atmospheric transmission and it is much more variable than over East Antarctica. Over the ocean, careful cloud and aerosol clearing would have to be done. This will be possible at night, but during the day it will be more difficult. However, over the ocean in daylight, clouds can easily be recognized over the dark ocean surface through their effect on the background. If cloud clearing can be done sufficiently well, then using the ocean surface return for calibration should be possible. In this section we present an algorithm for computing the system calibration coefficient using the surface return.

Repeating equation (2.1), the lidar equation can be written as:

$$S(z) = \frac{CE\beta(z)T^2(z)}{r^2} + p_b + p_d \quad (3.28)$$

Where $S(z)$ is the measured signal or photon count at height z , C is the system calibration coefficient, E is the laser pulse energy, $\beta(z)$ is the atmospheric backscatter coefficient at height z , and $T^2(z)$ the atmospheric transmission from the top of the atmosphere to height z , p_b is the solar background and p_d the detector dark count.

Following equation 2.15, the received number of photons can also be written as:

$$S(z) = \frac{P_e}{r^2} \beta(z) \Delta z A_t T^2(z) Q_e T_{opt} + p_b + p_d \quad (3.29)$$

Where P_e is the number of photons transmitted in the laser pulse. From equations 2.28 and 2.29 we can solve for C :

$$C = \frac{\lambda \Delta z A_t Q_e T_{opt}}{hc} \quad (3.30)$$

Since $P_e = E\lambda/hc$ where E is the laser pulse energy. The apparent surface reflectance (ASR) which is defined in greater detail in section 4.6 is:

$$\rho_{app} = \frac{\pi E_{rec} r^2 D_c}{EA_t T_{opt} Q_e} = \frac{\pi P_{rec} hcr^2}{\lambda EA_t T_{opt} Q_e} \quad (3.31)$$

Since the energy received from the surface return is $E_{rec} = \frac{P_{rec}hc}{\lambda}$ where P_{rec} is the number of photons received from the surface.

Substituting equation 2.29 into the right hand side of 2.31 yields:

$$\rho_{app} = \frac{\pi P_{rec} r^2 \Delta z}{CE} \quad (3.32)$$

Now, the measured apparent surface reflectance is simply the actual surface reflectance (R_s) times the two-way atmospheric transmission between the satellite and the ground (T_{atm}^2):

$$\rho_{app} = R_s T_{atm}^2 \quad (3.33)$$

Substituting 2.32 into 2.33 and solving for C gives:

$$C = \frac{\pi P_{rec} r^2 \Delta z}{ET_{atm}^2 R_s} \quad (3.34)$$

Thus, to compute the system calibration coefficient using ground returns, we must know the true surface reflectance (R_s), the number of photons received from the surface (P_{rec}), the emitted laser energy (E_{emit}), and the two-way atmospheric transmission between the satellite and the ground (T_{atm}^2). If we assume we can compute R_s over the ocean to within 5% (see section 4.7), one approach is to use equation 2.34 over large stretches of the ocean to create a scatter plot of C using 25 Hz data. For an assumed value of T_{atm}^2 the highest values of C will occur for the cleanest atmosphere. Averaging the top 10% or so of these values computed with T_{atm}^2 equal to the molecular value times ozone transmission may give reliable measurement of C. Alternatively, using data over the East Antarctic plateau, one can assume a surface reflectance value of 0.95 and an atmospheric transmission computed from molecular and ozone.

Table 3.7 Constants Used in this Section

Constant	Value
Ideal gas constant (R)	287.058 m ² s ⁻² °K ⁻¹
Gravity at sea level (g)	9.80171 ms ⁻²

Table 3.8. Algorithm adjustable parameters read in and used in this section.

Constant/Adjustable Parameter	Use/Meaning	Nominal Value
Cal_Cloud_Thresh	Threshold for excluding data in calibration zone for cal method 1	TBD

Cal_Latitude_Bound	The latitude boundary for calibration calculation for cal method 1	60 (N&S)
Cal_Integ_Time	Calibration integration time for cal method 1	75 sec
Cal_Scat_Ratio	Calibration Zone (13 to 11 km) aerosol scattering ratio	1.12
Cal_Atm_Trans	Particulate transmission from top of atmosphere to the calibration height (13 km) for cal method 1	0.95
Cal_Solar_Angle_Limit	Maximum solar elevation angle for calibration method 1 calculation	0.0

3.3.7.4 Calibration Error and Confidence

Because of the various limitations of the ICESat-2 atmospheric data and the fact that we likely will be able to compute only a few calibration points per orbit, the error and even confidence in calibration will be difficult to establish. We could estimate the error in calibration by computing the standard deviation of the calibration points over a given polar region or orbit, but we don't expect to have enough calibration points to make this plausible. Even if we could do this, we will not have calibration points outside of the polar regions and thus will not know if the system remains well calibrated there. A possible way to check on calibration outside of the polar regions would be to construct averages of cloud free (down to say 4-5 km altitude) calibrated, attenuated backscatter profiles and fit a line to this average profile. The absolute value of the difference between this line and the attenuated molecular backscatter profile summed (bin by bin) over the length of the profile could be used as a measure of calibration error. Most likely, this would only work over the nighttime hemisphere and would have to be done in the level 3ATL09 process.

For ATL04, we recommend a 3 scale system: 1 – little confidence, 2 – medium confidence, and 3 – high confidence. These levels will be based on the standard deviation of the computed calibration coefficient within a given orbit. The output parameter name for this is *Cal_Con* (in Table 3.1). The exact values of standard deviation to define the confidence levels will be determined in a later ATBD release. A more rigorous evaluation of calibration error is deferred to a future ATBD release.

3.4 Cloud Folding Flag

Clouds above 15 km will be folded down into the lower portion of the 14 km atmospheric profile. It is very beneficial to know when this is a possibility. Here we define the new parameter *Cloud_fold_flag* to be included on ATL09. This parameter is based on the GEOS-5 parameter CLDPRS or cloud top pressure, which is the pressure of the highest cloud layer in the current model grid box. We convert that pressure to a height and if it is greater than 15 km, the *Cloud_fold_flag* is set to 1. Otherwise it is 0. Rewriting Equation 2.3, we have:

$$z_2 = \frac{\overline{RT}}{g} \ln(P(z_1)/P(z_2)) + z_1 \quad (4.5a)$$

Where $P(Z_2)$ is the GEOS-5 parameter CLDPRS (cloud top pressure) and Z_2 is the geometric height of that pressure level. $P(Z_1)$ will be set to the surface pressure defined by GEOS-5 parameter PS and Z_1 will be the current surface height retrieved from the DEM. The average temperature of the layer (\bar{T} in °K) can be computed as the average of the temperature of all GEOS-5 layers between PS and CLDPRS, inclusive.\

3.4.1 Cloud Folding Flag Part II

While the above is useful, the GEOS-5 model's limited resolution and accuracy often results in missing clouds above 15 km that are actually present. For version 8.0 of this ATBD, the algorithm below adds other values to the cloud fold flag. The algorithm examines the signal magnitude of the bins below the ground to determine if the signal there is consistent with the magnitude of expected cloud signal. The algorithm below is designed to work with the raw photon count atmospheric profiles after background has been removed. The original cloud fold flag was only on the ATL09 product and was defined only by the presence of cloud above 15 km in the GEOS-5 model. The cloud fold flag should be computed and placed on the ATL04 product, since L3a software does not have access to the raw atmospheric profiles as does L2a. Note that *Cloud_fold_flag* must then be carried forward to the ATL09 product.

Note: in the below, we retain the value of *Cloud_fold_flag* as set by the method in section 3.4 (i.e. it is either zero or 1 at the beginning of the algorithm below). The value of *Cloud_fold_flag* is then modified by the below algorithm.

Inputs: The ATL04 parameter *surface_ht*, the vertically aligned, background subtracted raw profiles, DEM height, *cloud_fold_flag* from section 3.4, and the adjustable parameters in Table 3.10.

Output: *cloud_fold_flag*

Method:

1. If *surface_ht* is not invalid compute $hdiff = surface_ht - \text{height of the last valid bin of the profile}$. If *surface_ht* is invalid, then compute $hdiff = DEM_ht - \text{height of the last valid bin of the profile}$.
2. If *hdiff* is greater than $(fold_nbins+1) * bin_size$, then compute the mean of the last *fold_nbins* of the atmospheric raw profile. If *hdiff* is less than or equal to $(fold_nbins+1) * bin_size$, then compute the mean of the top *fold_nbins* of the profile (bins *va_first_bin* to *va_first_bin + fold_nbins*). The nominal value of *fold_nbins* is 8.
3. *Set_flag* = 0
4. If the *solar_elevation* is less than 0.0 and if the mean is $> fold_thresh_night$, then set *flag* = 1
5. If the *solar_elevation* is greater than or equal to 0.0 and if the mean is $> fold_thresh_day$, then set *flag* = 1

6. If flag eq 1 and if *Cloud_fold_flag* is already set to 1 (from the above MET based algorithm) then set *Cloud_fold_flag* = 3. If flag eq 1 and *Cloud_fold_flag* is zero, then set *Cloud_fold_flag* = 2
7. If flag eq 0, then do not alter the value of *Cloud_fold_flag*

Table 3.9

<i>Cloud_Fold_Flag</i> Value	Meaning
0	No folding was detected from the atm profile or indicated by the GEOS-5 model
1	Folding was indicated by the height of the highest cloud in the GEOS-5 model
2	Folding was indicated by the magnitude of the last few bins of the profile, but not by the GEOS-5 model
3	Folding was indicated by the magnitude of the last few bins of the profile AND by the GEOS-5 model

Table 3.10

Parameter	Nominal Value
fold_nbins	8
fold_thresh_night	5.0
fold_thresh_day	40.0

4 Level 3 Product (ATL09)

The atmosphere L3 product will be called ATL09 and will contain the calibrated, attenuated backscatter profiles and all other atmospheric parameters that can be obtained from the data. This will include cloud layer height, layer integrated attenuated backscatter and a host of other parameters discussed below. Note that the layer heights are discussed in a separate document: "ICESat-2 Algorithm Theoretical Basis Document for the Atmosphere, Part II: Detection of Atmospheric Layers and Surface Using a Density Dimension Algorithm". The output parameters listed in Table 4.1 with an asterisk are those that are described in Part II.

4.1 L3 Required Inputs

ATL04 of current granule and previous granule

NOAA Global Multi-sensor Snow/Ice Cover Map:

http://www.star.nesdis.noaa.gov/smcd/emb/snow/HTML/multisensor_global_snow_ice.html

Surface Albedo data set (global, 1x1 degree) by month – produced by science team (see section 4.6.2)

4.2 L3 Outputs

Currently, on ATL09 exists the “/profilex/high_rate/density” parameter which is dimensioned at (700,3). This parameter is described in the Atmosphere ATBD, part II and is a partial output of the Density Dimension Algorithm (DDA). The DDA can run in 2 modes: the first mode just creates one density field and the 2nd mode creates two density fields. The first density field is used to bring out the optically thick clouds and in the second density run, the data are aggregated with a larger kernel, to show the tenuous cloud layers and the aerosols. Over time, testing has determined that we will most likely be running the DDA in the “two density” mode. Because of this we request that the current “density” parameter be renamed to “density_pass1” and that a second parameter called “density_pass2” (also dimensioned at 700,3) be added to the product. If the DDA is ever run in the 1 density mode, then this parameter is populated with invalids.

Table 4.1. ATL09 Output Parameters

Parameter	Datatype	Units	Rate	Description	Source
Time	Float	seconds	25 Hz		
Latitude	Double	degrees	25 Hz	Based on range to local DEM	
Longitude	Double	degrees	25 Hz		
prof_dist_x	float	m	25 Hz	Along track distance from start of segment	ATL04
prof_dist_y	float	m	25 Hz	Across track distance from reference pair track	ATL04
CAB_Prof	Float(700,3)	$\text{m}^{-1} \text{sr}^{-1}$	25 Hz	Calibrated Attenuated Backscatter 3 strong beams from 20 to -1 km with vertical resolution of 30 m (eqn 3.4)	Computed
Mol_Att_Backscatter	Float(700)	$\text{m}^{-1} \text{sr}^{-1}$	1 Hz	Molecular Backscatter Profile 20 to -1 km (eqn 3.10)	ATL04
Mol_Backs_folded	Float(700)	$\text{m}^{-1} \text{sr}^{-1}$	1 Hz	Folded molecular backscatter profile (equation 3.17)	ATL04
MET_Temp	Float(700)	K	1 Hz	Temperature profiles from 20 to -1 km	ATL04
MET_Pres	Float(700)	Pa	1 Hz	Pressure profiles from 20 to -1 km	ATL04
MET_Latitude	Float	deg	1 Hz	Latitude of MET observation	ATL04

MET_Longitude	Float	deg	1 Hz	Longitude of MET observation	ATL04
MET_PS	Float	Pa	1 Hz	Surface Pressure	ATL04
MET_U2M	Float	ms ⁻¹	1 Hz	Eastward component of wind at 2m height	ATL04
MET_U10M	Float	ms ⁻¹	1 Hz	Eastward component of wind at 10m height	ATL04
MET_U50M	Float	ms ⁻¹	1 Hz	Eastward component of wind at 50m height	ATL04
MET_V2M	Float	ms ⁻¹	1 Hz	Northward component of wind at 2m height	ATL04
MET_V10M	Float	ms ⁻¹	1 Hz	Northward component of wind at 10m height	ATL04
MET_V50M	Float	ms ⁻¹	1 Hz	Northward component of wind at 50m height	ATL04
MET_T2M	Float	K	1Hz	Temperature at 2 m height	ATL04
MET_T10M	Float	K	1Hz	Temperature at 10 m height	ATL04
MET_QV2M	Float	Kg/Kg	1 Hz	Specific humidity at 2 m height	ATL04
MET_QV10M	Float	Kg/Kg	1 Hz	Specific humidity at 10 m height	ATL04
MET_TS	Float	K	1 Hz	Surface (skin) temperature	ATL04
MET_TROPPB	Float	Pa	1 Hz	Blended tropopause pressure	ATL04
MET_TROPT	Float	K	1 Hz	Tropopause temperature	ATL04
MET_TQL	Float	Kg/m ²	1 Hz	Total column cloud liquid water	ATL04
MET_TQI	Float	Kg/m ²	1 Hz	Total column cloud ice	ATL04
MET_CLDPRS	Float	Pa	1 Hz	Cloud top Pressure	ATL04
Backg_C	Float(3)	Photons per bin	25 Hz	Background used in computation of NRB (for each beam)	ATL04
Backg_Select	Integer	NA	1/granule	The background method used in calculation of NRB	ATL04
Backg_theoret	Float	Photons/bin	25 Hz	The theoretical background	Computed
Calib_Select	Integer	NA	1/granule	Used to select the calibration method. 1=cal method 1, 2=cal method2, 3=cal method 3	ATL04

Cal_C	Float(3)	Photons*m ³ sr/J	1 Hz	Calibration Coefficient (for each beam at 1 Hz)	Computed
Cal_Con	Integer(3)	NA	1/Orbit	Confidence in calibration	ATL04
Cal_Error	Float(3)		1/Orbit	Error in calibration	ATL04
Layer_flag		NA	25 Hz	Consolidated cloud flag	Computed
Layer_Top*	Float(10,3)	m	25 Hz	Height of top of detected layers (max 10)	Computed
Layer_Bot*	Float(10,3)	m	25 Hz	Height of bottom of detected layers (max 10)	Computed
Layer_Attr	Byte(10,3)	NA	25 Hz	Layer attribute flag for each layer	Computed
Cld_aer_discrim_thresh	Float	NA	NA	Adjustable threshold for determining whether a layer is cloud, aerosol or unknown. Nominal value of 0.25	Ancillary constant
Layer_Conf	Integer(10,3)	NA	25 Hz	Layer confidence flag for each layer	Computed
Layer_conf_dens	Float (10,3)	NA	25 Hz	Layer confidence flag for each layer based on density	Computed – See ATBD Part II
Cloud_Flag_Atm	Integer(3)	NA	25 Hz	Cloud flag from backscatter profile	Computed
Msw_Flag	Byte(3)	NA	25 Hz	Multiple Scattering warning flag	Computed
Cloud_Fold_Flag	Byte(3)	NA	25 Hz	Flag to tell user that this profile likely contains cloud signal folded down from above 15 km to the last 2-3 km of the profile. One flag for each profile. See Table 3.9 for flag value meanings	ATL04
Layer_IB	Float(10,3)	NA	25 Hz	Layer integrated backscatter	Computed
Column_OD_ASR	Float(3)	NA	25 Hz	Optical depth of atmosphere column from apparent surface reflectance	Computed
Column_OD_ASR_QF	Integer(3)	NA	25 Hz	Optical depth from ASR quality flag	Computed
Density_pass1*	Float(700,3)	NA	25 Hz	Density profile from the first density run to locate thick clouds	Computed
Density_pass2*	Float(700,3)	NA	25 Hz	Density profile from the second density	Computed

				run to locate thin clouds	
Layer_Dens*	Float(6,3)	NA	25 Hz	Integrated Layer Density	Computed
Surface_H_Dens*	Float(3)	m	25 Hz	Surface height from density	Computed
Bsnow_H_Dens*	Float(3)	m	25 Hz	Blowing snow layer height from density	Computed
Bsnow_Dens*	Float(3)	NA	25 Hz	Blowing snow layer integrated density	Computed
Ocean_Surf_Reflec	Float	NA	25 Hz	Ocean Surface Reflectance from Eqn 4.9	Computed
Apparent_Surf_Reflec	Float(3)	NA	25 Hz	Apparent Surface Reflectance from Eqn 4.7	Computed
Surface_Signal_Source	Byte	NA		Source of surface signal used to compute ASR	ANC
Surf_Refl_True	Float	NA	25 Hz	The value of $ACL_{R_{tru}}$ from section 4.6.2.2	Computed
Cloud_Flag_ASR	Integer(3)	NA	25 Hz	Cloud probability from ASR (Table 4.5)	Computed
Asr_cloud_probability	Integer	N/A	25 Hz	Probability of the occurrence of cloud based on the magnitude of the apparent surface reflectivity.	Computed
Phi_land	Float	NA	NA	Constant used in the computation of the threshold used to compute cloud probability from the ASR when over land	Ancillary constant
Phi_ocean	Float	NA	NA	Constant used in the computation of the threshold used to compute cloud probability from the ASR when over ocean	Ancillary constant
Bsnow_H	Float(3)	m	25,1 Hz	Blowing snow layer top height	Computed
Bsnow_OD	Float(3)	NA	25,1 Hz	Blowing snow layer optical depth	Computed
Bsnow_Con	Integer(3)	NA	25,1 Hz	Blowing snow confidence	Computed
Bsnow_PSC	Byte(3)	NA	25,1 Hz	Blowing Snow PSC flag	Computed
Bsnow_Prob	Float	NA	1 Hz	The probability of blowing snow	Computed

				occurrence based on MET data (Eqn 4.6a)	
Bsnow_intensity	Float	m/s	25,1 Hz	Blowing snow intensity defined as the average scattering ratio within the blowing snow layer times the 10 m wind speed.	Computed
Surface_height	Float(3)	m	25 Hz	Surface height from backscatter profile	ATL04
Surface_Signal	Float(3)	Photons	25 Hz	Surface signal	ATL04
Surface_Bin	Integer(3)	NA	25 Hz	Bin number of profile associated with the surface signal	ATL04
Surface_Width	Integer(3)	Bins	25 Hz	The number of bins comprising the surface signal for each beam. If > 1, the surface bin and height correspond to highest of those bins and the parameter Surface_Signal is sum of the signal value of those bins.	ATL04
Surface_Conf	Float	NA	25 Hz	The level of confidence in the surface signal magnitude and location (1.0 – lowest confidence; 100.0 – highest confidence)	ATL04
Surface_Thresh	Float				
Surf_type_igbp	integer	NA	25 Hz	IGBP Surface Type	ATL04
Dtime_select		NA		Control parameter for determining which method to use to compute the dead time correction factor	ATL04
Dtime_Fac2	Float(3)	NA	25 Hz	Dead time correction factor for surface signal computed from equation 2.25	ATL04
Dtime_Fac1	Float(3)	NA	25 Hz	Dead time correction factor for surface signal computed from radiometric lookup table	ATL04

ACLR_True	Float	NA	25 Hz	Clear sky initial ASR based on GOME climatology or Cox-Munk model: see Fig 3.6	Computed
USE_ATLAS	Integer	NA		Flag to control the computation of ACLR_True parameter: see pages 52-53	ANC
Msw_flag	Integer(3)	NA	1 Hz	Multiple scattering warning flag	Computed
Range_S	Float	km	25 Hz	Range from spacecraft to top of profile	ATL04
ATLAS_PA	Float	degrees	25 Hz	Pointing angle of spacecraft	ATL04
Laser_E	Float	Joules	10 Hz	Laser energy	ATL04
Solar_ZA	Float	degrees	1 Hz	Solar Zenith Angle	ATL04
Solar_AZ	Float	degrees	1 Hz	Solar Azimuth Angle	ATL04
Dem_h	Float	km	25 Hz	DEM value from a 1 kmx1km	ATL04
Surf_T	Integer	NA	1 Hz	IGBP Surface Type	ATL04
Snow_Ice	Integer	NA	1 HZ	NOAA snow-ice flag	ANC
Bckgrd_counts	Integer	Photons	200 Hz	Onboard 50 shot background	ATL03
Bckgrd_counts_reduced	Integer	photons	200 Hz	Onboard 50 shot background minus signal photons	ATL04
Bckgrd_rate	Float	Photons/s	200 Hz	50 shot background rate in per second (signal photons removed)	ATL04
Bckgrd_hist_top	Float	m	200 Hz	50 shot background integration window top height	ATL04
Bckgrd_int_height	Float	m	200 Hz	50 shot background integration window width	ATL04
Bckgrd_int_height_reduced	Float	m	200 Hz	bckgrd_int_height – signal photon height range	ATL04
Sig_count_hi	Integer(3)	photons	25 Hz	Ground signal photons for a 400 shot sum, high confidence	ATL04
Sig_count_med	Integer(3)	photons	25 Hz	Ground signal photons for a 400 shot sum, med confidence	ATL04
Sig_count_low	Integer(3)	photons		Ground signal photons for a 400	ATL04

				shot sum, low confidence	
Sig_h_mean_hi	Float(3)	m	25 Hz	Mean height of surface wrt EGM2008 ellipsoid, high confidence	ATL04
Sig_h_mean_med	Float(3)	m	25 Hz	Mean height of surface wrt EGM2008 ellipsoid, medium confidence	ATL04
Sig_h_mean_low	Float(3)	m	25 Hz	Mean height of surface wrt EGM2008 ellipsoid, low confidence	ATL04
Sig_h_sdev_hi	Float(3)	m	25 Hz	Standard deviation of high confidence surface signal photons	ATL04
Sig_h_sdev_med	Float(3)		25 Hz	Standard deviation of medium confidence surface signal photons	ATL04
Sig_h_sdev_low	Float(3)		25 Hz	Standard deviation of low confidence surface signal photons	ATL04
Podppd_flag	byte	NA	25 Hz	Precision orbit and pointing flag from ATL03	Podppd_flag

***Described in ATBD Part II. Note Data product quality parameters not listed.**

Table 4.2. Partial list of algorithm adjustable parameters read in and used by the ATL09 algorithm.

Constant/Adjustable Parameter	Use/Meaning	Nominal Value
BS_Thres_wind	Surface wind speed minimum in order for blowing snow to exist	4.0
BS_Thresh_Scale	Scaling factor for computing the blowing snow threshold (molecular*BS_Thresh_Scale)	10.0
BS_Top_Scale	Scale factor to obtain top of layer threshold (BS_Top_Scale*	1.0

BS_Extinc_Backs	The assumed blowing extinction to backscatter ratio	25.0
Lr_bsnow_fac	Scaling factor for blowing snow threshold used for the low rate blowing snow detection	0.5
max_bsnow_cab	Maximum attenuated backscatter allowed for blowing snow detection	4.0e-4
Cal_Cloud_Thresh	Threshold for excluding data in calibration zone (method 1)	2.5E15
Cal_Latitude_Bound	The latitude boundary for calibration calculation	60 (N&S)
Cal_Back_Max	The background level above which calibration will not be calculated (summed)	0.60 MHz
Cal_Integ_Time	Calibration integration time	75 sec
Cal_Scat_Ratio	Calibration Zone (13 to 11 km) aerosol scattering ratio	1.05
Cal_Atm_Trans	Particulate transmission from top of atmosphere to the calibration height (13 km)	0.95
Cal_Solar_Angle_Limit	Minimum solar zenith angle for calibration calculation	0.0
Use_ATLAS	0/1 flag to tell cloud detection using apparent surface reflectance algorithm to use ATLAS data	0
Cal_Default	Calibration constant default value if it cannot be calculated from the data	1.5E21
Cal_Min	Calibration constant minimum limit	5.0E20
Cal_Max	Calibration constant maximum limit	2.5E21
Calib_select	Flag to tell user which calibration method was used to produce the CAB profiles. 1=cal method 1, 2=cal method2, 3=cal method 3	3

Surface_Signal_Source	1/2 flag to tell ASR algorithm which surface signal to use. 1 = from ATL04; 2 = from ATL03	1
-----------------------	--	---

4.3 Calibrated, Attenuated Backscatter Profiles

The ATL04 processing has generated the normalized relative backscatter profiles as:

$$NRB'(z) = (S'(z) - p_b - p_d)r^2 / E = C\beta(z)T^2(z) \quad 4.1$$

Where $S'(z)$ is the raw photon count profile corrected for molecular folding. The calibrated, attenuated backscatter profiles are obtained by simply dividing the NRB profiles by the calibration coefficient (C) which was computed in ATL04:

$$\beta(z)T^2(z) = (\beta_p(z) + \beta_m(z))T_p^2(z)T_m^2(z) = NRB'(z) / C \quad 4.2$$

For calibration method 1, the calibration coefficient is computed only over the polar regions and stored in ATL04, most likely producing only 3-4 calibration values per orbit. After launch it was decided that calibration method 1 was untenable. There is no way that method can work. For ASAS release 5.2, data product version 002, calibration method 2 was used. For ASAS release 5.3, data product version 003, calibration method 3 will be used. Calibration method 3 computes ~60 calibration constants continuously around the orbit, independently for each granule. This means that there can be a slight calibration ‘jump’ from the end of a granule to the beginning of the next granule. However, we do not anticipate this to be large or noticeable. To obtain a calibration value at any time t within a granule, a linear piece-wise interpolation between calibration points will be used. All profiles that occur at and between the first and last calibration values will use a linear least squares fit to the calibration points in the current granule to assign a calibration value to time t within the granule:

$$C(t) = a + bt \quad 4.3$$

Where t spans the entire current granule time. The calibration value for profiles before the first calibration point of the granule will be set to that first calibration value and the calibration value for profiles after the last calibration value of the granule will be set to the last calibration value.

The calibrated attenuated backscatter computation is then a function of time:

$$\beta(z,t)T^2(z,t) = NRB'(z,t) / C(t) \quad 4.4$$

The calculated calibration coefficient should be calculated at a time resolution of 1 second and is also on the ATL09 product at that resolution (parameter *Cal_C*).

4.4 Layer Heights and Flags from Backscatter Profiles

The detection of layer heights will be performed on each of the 3 strong beams using the Density Dimension Algorithm (DDA) that is detailed in Part II of the Atmospheric ATBD. Please refer to that document for information pertaining to the algorithm. Here we define the parameters associated with the layers retrieved by the DDA and related flags.

There will be a maximum of ten layer heights (top and bottom) stored on the product for each of the 3 strong beams – parameters *layer_top* and *layer_bot*. Associated with these will be a layer confidence flag (parameter *layer_conf*) and a layer attribute flag (parameter *layer_attr*). The confidence flag is calculated by averaging the calibrated backscatter within a layer and dividing it by the average molecular backscatter for the same vertical range. The integer value of this calculation will constitute the confidence flag (a value from 0 to approximately 1000). If this ratio is less than unity, it will be set to 0. The higher the number, the greater the confidence.

The layer attribute flag is intended to discriminate between cloud and aerosol. If the layer top is above 6 km, then the layer is cloud (*layer_attr* = 1). If the layer top is at or below 6 km and the layer confidence is less than 10, then the layer is aerosol (*layer_attr* = 2). If the layer top is less than 6 km, and the layer confidence is greater than or equal to 20, then the layer is cloud (*layer_attr* = 1). If the layer top is less than 6 km and the layer confidence is greater than or equal to 10 and less than or equal to 20, then the layer type is unknown (*layer_attr* = 3).

Also included on the product is a cloud flag which can be used to determine whether or not a layer was detected for a given profile (ATL09 product parameter *Cloud_Flag_Atm*). The *Cloud_Flag_Atm* flag is set to a positive number corresponding to the number of layers found. This flag should be used for nighttime (solar zenith angle > 90.0) data as solar background will make the detection of clouds very difficult and prone to false positives. For daytime data it is best to use the ATL09 product parameter *Cloud_Flag_ASF* which is based on Apparent Surface Reflectance (please see section 4.6.2 and Table 4.5 for further information on this flag).

Atmospheric layers such as clouds and blowing snow can cause multiple scattering which will increase photon path length and make a surface appear to be lower than it actually is. The magnitude of multiple scattering is related to the height and optical depth of the scattering layer. The lower and denser the layer, the greater the multiple scattering. Hence, thick blowing snow or fog layers (which touch the ground) are of greatest concern. In theory it is possible to calculate the magnitude of photon delay if the height, thickness, optical depth and particle size are known. Unfortunately, we will not be able to measure these parameters from the ATLAS data itself accurately enough to compute a delay. Instead we use the height, thickness and estimated optical depth of the layer to produce a multiple scattering warning flag. Note that these parameters are derived from the atmospheric backscatter profiles and are only viable for nighttime data.

The multiple scattering warning flag (ATL09 parameter *msw_flag*) has values from -1 to 5 where zero means no multiple scattering and 5 the greatest. If no layers were detected (meaning no cloud layers or blowing snow), then *msw_flag* = 0. If blowing snow is detected and its estimated optical depth is greater than or equal to 0.5, then *msw_flag* = 5. If the blowing snow optical depth is less than 0.5, then *msw_flag* = 4. If no blowing snow is detected but there are cloud or aerosol layers detected, the *msw_flag* assumes values of 1 to 3 based on the height of the bottom of the lowest layer: < 1 km, *msw_flag* = 3; 1-3 km, *msw_flag* = 2; > 3km, *msw_flag* = 1. A value of -1 indicates that the signal to noise of the data was too low to reliably ascertain the presence of cloud or blowing snow. We expect values of -1 to occur only during daylight. The algorithm to determine when this condition is present is TBD.

4.4.1 Layer Integrated Attenuated Backscatter

While it is highly desirable to compute the optical depth of the cloud and aerosol layers that are detected by the layer finding algorithm (discussed in Part II of the ATBD – a separate document), because of the various limitations of the atmospheric channel, this may prove to be impossible. The main problem is that we need to know the total attenuation of the signal from the top of the atmosphere down to the top of the layer. In many (most) cases we will not know this. There may be cases where the atmosphere is totally clear down to the first layer at say 13 km, but we can never be sure of that. Remember, we are totally blind in the 14-15 km region and the scattering above that is folded down into the lower few km of the atmosphere. In essence we are screwed. Possibly, over the polar regions, optical depth retrieval would be possible, but until we get actual ATLAS data we do not feel it is worth pursuing here. Instead we will compute and store on the ATL09 product the layer integrated total (particulate plus molecular) attenuated backscatter, which is related to optical depth. The integrated backscatter is thus defined as

$$\gamma' = \int_{z_{top}}^{z_{bot}} (\beta_p(z) + \beta_m(z)) T_p^2(z) T_m^2(z) dz \quad (4.5)$$

Where z_{top} is the top of the layer and z_{bot} the bottom. At this point we recognize that $\beta_p(z)$ can have particulate scattering from $z+15$ km (as will $\beta_m(z)$, but it will be small). However, above 4 km, this should in general not be the case. When layers occur below 4 km, we cannot guarantee that the particulate scattering is actually from that height. In these cases, a quality flag will be set to indicate this. There will be an integrated backscatter value for each layer detected (maximum of 6). Note that the integrand in equation 3.5 is simply the calibrated, attenuated backscatter as in equation 3.2.

4.5 Cloud/Aerosol Discrimination

4.5.1 Background

ATLAS atmospheric lidar data provides the information necessary to locate layers of particulates within the observed atmospheric volume. It is also capable of providing an informed judgement of the species of the observed particles. The layers are considered to be either clouds, where the

particles are condensed water either liquid or ice, or aerosols, where the particles are other liquid or solid substances that do not undergo phase changes in the atmosphere. Two examples of clouds are high altitude cirrus and low altitude marine stratocumulus. Two examples of aerosols are marine boundary layer haze and smoke from agricultural burning. Analysis of the atmosphere relies on the knowledge of the presence of clouds or aerosols so it is important to provide discrimination in the data products.

The strength of the lidar signal is the main quantity in the ATLAS atmospheric channel used for layer type discrimination. The strength of the signal is related to the particle attenuated backscatter coefficient, $\beta'(r)$, where r is the range between the instrument and the scattering particle. Cloud particles tend to have a significantly larger $\beta'(r)$ than aerosol particles., but not all the time. So, the magnitude of the lidar signal can be used as one parameter to discriminate between aerosol and cloud layers. The characteristic difference in $\beta'(r)$ magnitudes is related to the altitude of the layers. Low clouds will typically have $\beta'(r)$ much larger than low aerosols but magnitudes of high clouds will be lower and closer to that of aerosols. Therefore, altitude of a layer needs to be included in the discrimination of cloud and aerosols when using $\beta'(r)$ as a distinguishing parameter.

Relative humidity, RH, provides additional information to the cloud/aerosol discrimination process. Clouds will tend to exist only where the RH is high, approaching 100%. Aerosols can exist in both high and low relative humidity environments. Coincidental RH is available in the atmospheric model information retrieved to coincide with the ATLAS data.

Based on the considerations stated above, the ATLAS cloud/aerosol discrimination algorithm will use attenuated backscatter coefficient, layer altitude, and relative humidity to distinguish between clouds and aerosols. The technique will be statistical in nature where empirically derived tables are used to produce a confidence factor for a specification of cloud or aerosol.

4.5.2 Layer Discrimination Confidence Factor Algorithm

The ATLAS layer discrimination algorithm is adapted from a method developed for use for CALIOP lidar which is on the CALYPSO spacecraft. The CALIOP algorithm is described in the publication ***Use of probability distribution functions for discriminating between cloud and aerosol in lidar backscatter data*** Zhaoyan Liu, MarkVaughan, David M. Winker Chris A. Hostetler Lamont R. Poole, Dennis Hlavka, William Hart, Matt McGill; Journal of Geophysical Research: Atmospheres / Volume 109, Issue D15; 07 August 2004, <https://doi.org/10.1029/2004JD004732>.

The discrimination algorithm for ATLAS uses a confidence function to quantify the likelihood that a layer is comprised of either cloud or aerosol particles. The confidence function is a linear combination probability distribution functions (PDF) of clouds and aerosols. The PDFs are found for a two dimensional domain since $\beta'(r)$ and RH are the physical quantities used to discern the

particle species. Since altitude has a strong influence on the PDF, confidence functions are derived for a set of altitude ranges.

The 2-D PDF's are empirically derived from ATLAS data for finite intervals of $x=\log_{10}[\beta'(r)]$ and $y=RH$. Logarithms of $\beta'(r)$ are used since the pertinent values extend over several orders of magnitude. For a particle species i , the probability that the parameters are contained within a neighborhood $x\pm Dx, y\pm Dy$ is given by

$$p_i(x\pm Dx, y\pm Dy) = n_i(x\pm Dx, y\pm Dy) / N_i \quad (4.5.1)$$

where n_i is the number of occurrences within the xy boundaries and N_i is the total number of observations of species i . The probability density is

$$P_i(x\pm Dx, y\pm Dy) = p_i(x\pm Dx, y\pm Dy) / Dx Dy = [n_i(x\pm Dx, y\pm Dy) / N_i] / Dx Dy \quad (4.5.2)$$

So,

$$n_i(x\pm Dx, y\pm Dy) = P_i(x\pm Dx, y\pm Dy) N_i Dx Dy \quad (4.5.3)$$

For two species, 1 and 2, which pertain to cloud and aerosol particles, the two dimensional confidence function is defined as

$$f(x, y) = [n_2(x\pm Dx, y\pm Dy) - n_1(x\pm Dx, y\pm Dy)] / [n_2(x\pm Dx, y\pm Dy) + n_1(x\pm Dx, y\pm Dy)] =$$

$$[P_2(x, y)N_2 - P_1(x, y)N_1] / [P_2(x, y)N_2 + P_1(x, y)N_1] =$$

$$[P_2(x, y) - P_1(x, y)K_s] / [P_2(x, y) + P_1(x, y)K_s] \quad (4.5.4)$$

where $K_s = N_1/N_2$.

The probability density functions $P_i(x, y)$ are determined from the data cases where the sampling space needs to represent the universe of values of the parameters $\beta'(r)$ and RH . A practical difficulty in confidence function development is locating enough samples where the species are known and segregated to represent the required 2 dimensional domain size. The difficulty is addressed by having an analyst find many cases to develop reliable confidence functions. This laborious process is independent of the data processing. The results are an input into the processing algorithm

4.5.3 Implementation of the Layer Discrimination Algorithm

Prior to production runs, the confidence values for the expected 2-D domain are determined from existing ATLAS data. These are stored in a table that is constructed for input into the analysis algorithms. Many granules are analyzed to find regions where the particle type is known. Probability density functions, $P_1(x, y)$ for aerosols and $P_2(x, y)$ for clouds, of the particle characteristics and environment are computed for many selected samples. By using equation 4.5.4, $P_1(x, y)$ and $P_2(x, y)$ are used to derive a

confidence values, $f(x,y)$, for sets of parameter intervals. The confidence values are stored in such a way that they can be indexed according to the cloud and environment characteristics.

Layer altitude has an influence on layer parameter statistics so ATLAS confidence values are computed for a set of layer altitude ranges. In each altitude range, confidence values are computed in a 2 dimensional domain. The dimension parameters are $\log_{10}(\beta'(r)/\beta'_0)$ and RH. The parameter β'_0 is the minimum of the β' range. The confidence values are computed for interval ranges of $\log_{10}(\beta'(r)/\beta'_0)$ and RH. The magnitude of range for each of these intervals is constant so that an index can be computed to retrieve a confidence value.

The analysis software will read the confidence table and retain it in memory. The current table constructed for the ATLAS data product has 5 altitude intervals, 15 $\log_{10}(\beta'(r)/\beta'_0)$ intervals, and 4 RH intervals. The altitude intervals are in meters [-1000,500], [500,1000], [1000,5000], [5000,8000], and [8000,20000]. Since these intervals are not constant magnitude, indexing cannot be used. The 15 equal $\log_{10}(\beta'(r)/\beta'_0)$ intervals are computed for the range $1 \times 10^{-7}/\text{m-sr}$ to $1 \times 10^{-4}/\text{m-sr}$ with $\beta'_0 = 1 \times 10^{-7}/\text{m-sr}$. The 4 equal RH intervals are computed for the range 0-100% yielding [0,25], [25,50], [50,75], and [75,100].

NOTE: This table has been constructed by the authors and supplied to the ASAS team. The ASAS codes just need to read the table in, store it in memory, and compute the indices used to index into the table as described below.

Computer code to read the table into memory follows these steps.

Open file

Read in number of altitude intervals, number of $\log_{10}(\beta'(r)/\beta'_0)$ intervals, number of RH intervals

For each of the 5 altitude intervals, read in

Altitude range boundaries

For each of 4 RH interval read in

RH range boundaries and confidence values for each of 15 $\log_{10}(\beta'(r)/\beta'_0)$ intervals

The values of β' for the lower boundary of the 15 intervals

Close file

IDL statements to read the table used in development are:

`nzrr=0L ;initialize variable number of altitude levels`

`nabcbinsr=0L ;initialize variable number of β' bins`

```

nrhbinsr=0L           ;initialize variable number of RH bins

conftablefile='/home/billhart/IcesatII/cloudaer/idl/layers/conftable.txt'           ;confidence table file

openr,conlun,conftablefile,/get_lun           ;open file

readf,conlun,nzrr,nabcbinsr,nrhbinsr,format='(3i8)' ;number of bins

nzc=nzrr           ;store number of altitude levels

nabcbins=nabcbinsr           ;store number of ' ' bin

nrhbins=nrhbinsr           ;store number of RH bins

zrbounds=make_array(nzc,/double,value=0.0d0);altitude bounds

rhbounds=make_array(nrhbins,/long,value=0L);RH bounds

conftable=make_array(nabcbins,nrhbins,nzc,/double,value=0.0d0);

;confidence table

zc=0.0d0           ;initialize altitude variable

abcr=make_array(nabcbins,/double,value=0.0d0);initial ' ' variable

rhr=[0L,0L]           ;initialize relative humidity variable

ctr=make_array(nabcbins,/double,value=0.0d0);table variable

for i=0L,nzc-1L do begin           ;loop over altitude ranges

    readf,conlun,zc           ;read altitude boundary

    zrbounds(i)=zc           ;store altitude boundary

    for j=0L,nrhbins-1L do begin ;loop over RH bins

        readf,conlun,rhr,ctr           ;read RH and confidence vector

        readf,conlun,abcr           ;read ' ' bin boundaries

        if(i eq 0L)then begin           ;first pass in altitude loop

            rhbounds(j)=rhr(0)           ;store RH boundary

        endif

        conftable(*,j,i)=ctr           ;store confidence for altitude and RH

    endfor

endfor

```

```

if(i eq 0L)then abcbounds=abcr ;store  $\beta'$  bin boundaries

endfor                                ;end read loop

free_lun,conlun                        ;confidence table read complete

```

Vectors of the bin boundaries for each of the indices are read from the table. See zrbounds, rhbounds, and abcbounds in the sample code. Adjacent pairs of elements in each vector form the range of values for the index into the confidence table. The index is the lower value of the bin range.

Once the table is stored in memory, the confidence factor for each of the high resolution layers for each of the ATLAS profiles can be determined from the layer parameters. In the sample code above, the confidence factors are stored in a 3-D array, conftable(15,4,5). The first dimension is for the β' index, the second is for the RH index, and the third is for altitude. For each layer found at high resolution, the indexes are determined with the following procedures.

The average $\beta'(r)$, altitude, and RH is computed for the layer. Since the RH is available only at low resolution, a computation is required to co-locate the high resolution observations with the low resolution observations. Then the index of the range of the boundary vectors that contains

For each high resolution layer:

Compute \underline{Z} mean altitude of the layer

Compute $\underline{\beta'}$ mean attenuated backscatter coefficient of layer

Compute \underline{RH} mean relative humidity of layer

Determine indices i, j, k where

i: $Z_i \leq \underline{Z} < Z_{i+1}$

J: $\beta'_j \leq \underline{\beta'} < \beta'_{j+1}$

K: $RH_k \leq \underline{RH} < RH_{k+1}$

Retrieve confidence for the layer

$C_i = C(i, j, k)$

$|C_i| < \chi$

T

Layer type indeterminate

F

F

Layer type cloud

$C_i > 0$

T

Layer type aerosol

each of the averages is determined for each parameter. The index of the range is the lower of the adjacent boundary vector elements. The three indices are used to retrieve the confidence value for the layer from the confidence table. The confidence table is designed so the values greater than 0 represent aerosols and values less than 0 represent clouds. The greater the magnitude of the value, the stronger the confidence in the results.

The flow chart above shows the progression of the calculation of the confidence table indices. In the chart, z_j , β'_j , and RH_k are elements of the layer altitude, attenuated backscatter coefficient, and relative humidity bin boundary vectors read in with the confidence table. The threshold level shown in the flow chart (χ) will be an adjustable parameter on ATL09 with the initial value of 0.25.

The IDL code below finds the confidence factor for each layer in a data granule. It is called for each of the 3 layer profiles. The confidence table contable, which has been read in earlier, holds the cloud/aerosol confidence factors as a function of attenuated backscatter coefficient(cab) and relative humidity(rh) for several altitude intervals. For each layer, indices into contable are determine from the mean altitude, mean attenuated backscatter coefficient, and mean relative humidity of the layer.

```

pro gettypecloudaero,lathr,zrbounds,abcbounds,rhbounds,conftable,$
    deltatime,deltatimelr,ATLASsdpgpsepochn,binsize,frametop,$
    ltp,lbt,cab,rh,nl,lattr,layerconfidence

nolay=3.0e+38

nhr=n_elements(cab(0,*))      ;nhr=number high resolution obs

nplay=n_elements(ltp(*,0))    ;nplay=number of potential layers in a profile

layerconfidence=make_array(nplay,nhr,/float,value=-9.9);output confidence

nzbounds=n_elements(zrbounds) ;zrbounds-> altitude bin boundaries

nrhbounds=n_elements(rhbounds);rhbounds->RH bin boundaries

nabcbounds=n_elements(abcbounds);abcbounds->b' bin boundaries

for i=0L,nhr-1L do begin      ;loop through all high resolution profiles

    dtim=deltatime(i)         ;high resolution time

    timediffs1hz=abs(deltatimelr-dtim) ;deltatimelr->low resolution time vector

    td1hzn=min(timediffs1hz,idxtdn) ;td1hzn=min time difference hr to lr times

    ilr=idxtdn                 ;ilr=low resolution index at min time difference

```

```

ltops=reform(ltp(*,i))      ;layer tops for profile i

lbots=reform(lbt(*,i))      ;layer bots for profile i

idxnl=where(ltops gt nolay)  ;layer indexes of valid layers

nlayers=10L;this method was installed because of atl09 granule
04jan19,01:08:52;19aug19;profile=54633

if(idxnl(0) gt -1L)then begin

    nlayers=idxnl(0L)        ;nlayers=number of valid layers in profile i

endif

for j=0L,nlayers-1 do begin  ;loop through layers if nlayers>0

    bintop=long((frametop-ltops(j))/binsize);index of bin top

    binbot=long((frametop-lbots(j))/binsize);index of bin bottom

    if(binbot lt bintop)then binbot=bintop ;ensure valid bin indices

    nbinsl=binbot-bintop+1L      ;nbinsl=number of bins in layer

    abctotl=0.0

    for k=bintop,binbot do begin ;sum over layer bins

        abctotl+=cab(k,i)

    endfor

    abclaymean=abctotl/float(nbinsl) ;compute mean b' of layer

    altlaymean=(ltops(j)+lbots(j))/2.0 ;compute mean altitude of layer

    rhtotl=0.0

    for k=bintop,binbot do begin;sum over layer bins

        rhtotl+=rh(k,ilr)

    endfor

    rhlaymean=rhtotl/float(nbinsl) ;compute mean RH of layer

    idxzr=nzrbounds-1L

    idxz=where(altlaymean lt zrbounds)

```

```

if(idxz(0) gt -1L)then idxzr=idxz(0)-1L;idxzr=index of altitude regions

idxrh=nrhbounds-1L

idxr=where(rhlaymean lt rhbounds)

if(idxr(0) gt -1L)then idxrh=idxr(0)-1L ;idxrh=index of RH

idxab=nabcbounds-1L

idxa=where(abclaymean lt abcbounds)

if(idxa(0) gt -1L)then idxab=idxa(0)-1L

idxab=max([idxab,0L]) ;idxab=index of b'

confidence=conftable(idxab,idxrh,idxzr);retrieve layer confidence

layerconfidence(j,i)=confidence ;store layer confidence in hr array

endfor

endfor

return

end

```

Upon exit from the procedure, the array `layerconfidence` holds the confidence value for each layer in each lidar profile. If the magnitude of confidence is greater than χ (nominal value of 0.25), the layer attribute is deemed aerosol but if it is less than $-\chi$, the layer attribute is deemed cloud. If the magnitude is less than χ and greater than $-\chi$, the layer attribute is deemed indeterminate. On the ATL09 data product χ will be an adjustable constant and be called *cld_aer_discrim_thresh*.

On the ATL09 product, the parameter which identifies cloud or aerosol is called `layer_attr`. Thus if the magnitude of the computed confidence is less than $-\chi$, then set *layer_attr*=1 (cloud). If the magnitude of the computed confidence is greater than χ , then set *layer_attr*=2 (aerosol). If the magnitude of the computed confidence is less than χ and greater than $-\chi$, then set *layer_attr*=3 (unknown).

4.6 Blowing Snow

4.6.1 Blowing Snow Layer Height

The blowing snow threshold is constructed from a scaling factor (*bs_thresh_scale*) times the magnitude of 532 nm attenuated molecular (Rayleigh) scattering at the surface height of the current retrieval location times a factor (*bsnow_sf*) as in equation 4.6.1 below. The value of *bsnow_sf* depends on whether the blowing snow detection algorithm is being used for full (25

Hz) resolution detection, or low resolution (1 second) detection. If the blowing snow detection algorithm is being used with 25 Hz profiles, $bsnow_fac=1.0$. If the algorithm is being used with 1 second average profiles, $bsnow_fac = lr_bsnow_fac$, where lr_bsnow_fac is an adjustable constant with the nominal value of 0.5.

$$T = \beta_m * bs_thres_scale * bsnow_sfac \quad (4.6.1)$$

The scaling factor (*BS_Thresh_Scale*) has a nominal value of 10.0 and was determined by an iterative approach of adjustment and review of retrieval results until they were satisfactory. The resulting threshold must be great enough to insure minimal false positive detections while small enough to retain adequate sensitivity.

The blowing snow module will output 4 parameters stored on ATL09 (see Table 4.1): high rate and low rate *Bsnow_H*, *Bsnow_OD*, *Bsnow_Con*, *Bsnow_intensity* and *Bsnow_PSC* which are blowing snow height (height above the surface of the top of the blowing snow layer), optical depth of the layer, blowing snow confidence flag, blowing snow intensity (the value of chi from equation 4.6.3) and blowing snow PSC flag, respectively. The blowing snow detection algorithm will be invoked over any surface determined to be snow, ice sheet or sea ice. This determination is made by reading in the NOAA daily snow/ice cover map (see section 4.7.2.2 and Figure 4.7.8). If the orbit is not over snow, ice sheet or sea ice, then both the high rate and low rate ***Bsnow_H*, *Bsnow_OD* and *Bsnow_Con* should be set to invalid.**

If over the appropriate surface type, the algorithm must first check to see whether the surface was detected. This can be done by checking the ATL04 high rate parameter “Surface_Bin”. If Surface_Bin is not invalid, then the surface has been detected.

The below gives the steps of the blowing snow algorithm.

1. **Is the current position of the orbit over snow, ice sheet or sea ice? If yes, go to step 2. If no,** then set the high rate and low rate ***Bsnow_H*, *Bsnow_OD* and *Bsnow_Con* to invalid and go to step 7.**
2. If the surface has been detected and the 10 m wind speed is greater than BS_Thresh_Wind (nominally 4 m s⁻¹ see Table 4.2), then continue to step 3. If the surface is not detected then **set *Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to -3 and go to step 7.** If the surface is detected, but the wind speed is less than BS_Thresh_Wind then **set *Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to -2 and go to step 7**
3. Check the backscatter signal in the bin immediately above the ground (Surface_Bin-1) to see if it exceeds the blowing snow threshold (T in eqn 4.6.1, about $2.0 \times 10^{-5} \text{ m}^{-1} \text{ sr}^{-1}$) AND the value of the attenuated backscatter of that bin ($cab_profile(surface_bin-1)$) is less than or equal to max_bsnow_cab (nominal value of 4.0×10^{-4}). If it is, set “start_bin” to surface_bin-1 and then continue to step 5 below.

4. If the value of the backscatter immediately above the ground (surface_bin-1) is greater than *max_bsnow_cab*, then check the value of the backscatter of surface_bin-2. If that is \geq *bsnow_thresh* and \leq *max_bsnow_cab* then set “start_bin” = surface_bin-2 and continue below. If it is not, we have not detected blowing snow. **set *Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to -1.** go to step 7.
5. Interrogate bins from start_bin-2, start_bin-3, etc., moving upward until the backscatter within the bin is less than the blowing snow threshold times a scale factor (*BS_Top_Scale*) that is nominally 1.0. Call this bin T. The top of the blowing snow layer is then the bin immediately below bin T (i.e. bin T+1). The search for the top will not exceed 500 m above the ground as the likelihood of a blowing snow layer deeper than this is very small. If the search upward has reached 500 m, but no top of the layer was found, then **set *Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to 0.**
6. Once the top of the layer has been found (at bin T), the height of the blowing snow layer (parameter *Bsnow_H*) is then defined as (start_bin-T+1) * 30.0, where T is the layer top bin and start_bin is defined in step 2 or 3. *Bsnow_OD* and *Bsnow_Con* are then defined following the discussion in sections 4.6.2 and 4.6.3, respectively.
7. Go to next profile

Table 4.2.1

Blowing Snow Algorithm Control Parameter	Value
<i>Bs_thresh_scale</i>	10.0
<i>Lr_bsnow_fac</i>	0.5
<i>Bs_top_scale</i>	1.0
<i>Max_bsnow_cab</i>	4.0e-4

Note that this algorithm will be applied to the full resolution atmospheric profiles (to produce the *high_rate* blowing snow height) and to profiles that are averaged to 1 second to produce a *low_rate* blowing snow height parameter. In the case of the low rate blowing snow height, the location of the ground bin should be determined as the highest bin of the valid surface bins that make up that second of data. If there are less than 10 valid surface bins for this second, then set the **low rate *Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to -3.** If there are \geq 10 valid surface bins for this second, but the wind speed is less than *BS_Thresh_Wind* then **set *Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to -2.** If blowing snow is not detected, **set *Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to -1.** If the top of the layer cannot be found, then set ***Bsnow_H* and *Bsnow_OD* to invalid and *Bsnow_Con* to 0.**

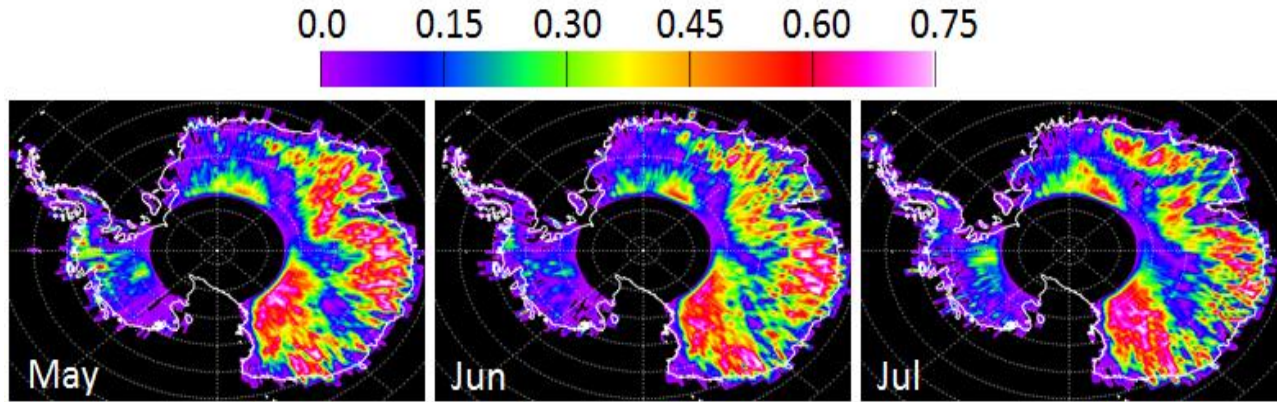


Figure 4.6.1 Blowing snow frequency over Antarctica for May-July, 2009 as derived from analysis of CALIPSO data.

Table 4.3

Value of <i>Bsnow_con</i>	Meaning
invalid	Orbit not over land ice, snow or sea ice
-3	Surface not detected
-2	Surface detected, but the 10 m wind speed is less than <i>BS_Thresh_Wind</i>
-1	Blowing snow was searched for, but not found
0	Blowing snow layer top not found
> 0	Blowing snow was found. See Section 4.6.3 for the computation of the value of <i>Bsnow_con</i>

4.6.2 Blowing Snow Layer Optical Depth

The optical depth of the layer is then computed using an assumed extinction to backscatter ratio. (*BS_Extinc_Backs*) with a nominal value of 25 sr (see Table 4.2). To compute the blowing snow layer optical depth (τ_{bs}), we sum the calibrated attenuated backscatter within the blowing snow layer and multiply by the product of the bin size and the extinction to backscatter ratio (*S*):

$$\tau_{bs} = B_s S \sum_{z=z1}^{z2} \beta(z) \quad (4.6.2)$$

Where *z1* is the top bin of the layer (bin *T*) and *z2* the bottom bin (bin *start_bin*), *B_s* is the bin size in meters (30), and $\beta(z)$ is the calibrated attenuated backscatter.

The extinction to backscatter ratio of blowing snow is not known and is a subject of current research by the author. Algorithm details and further example of retrievals can be found in Palm et al., 2011.

The output of the blowing snow detection algorithm has been extensively checked for consistency and quality by generating and reviewing hundreds of images of the detected blowing snow layers using both ICESat-1 and CALIPSO data. A limitation of the lidar technique is that the blowing snow layer has to be at least 30 m thick in order for enough backscatter signal to be collected in the bin immediately above the ground. This means that shallow blowing snow layers, which may be frequent, will probably not be detected. Further, blowing snow cannot be detected beneath thick or highly attenuating layers (tropospheric or polar stratospheric clouds with optical depth > about 2.5-3.0), since detection of a ground return is required. The latter limitation implies that most of the blowing snow associated with winter storms (cyclones) will go undetected. These limitations will certainly result in lower blowing snow frequencies than actually exist. Furthermore, the magnitude of the discrepancy will depend on the cloud cover frequency of a given region. For instance, along the coast of Antarctica where blowing snow frequency is known to be high, it is also cloudier than more inland regions.

We also recognize that scattering from the region 15 km above the surface that is folded into the near-surface scattering can produce a false positive detection. This would only happen in the presence of polar stratospheric clouds that occur most frequently in late winter and early spring in high latitudes. We have a flag on the ATL09 produce (*Bsnow_PSC*) which indicates the potential of PSCs to affect the blowing snow retrieval. The flag will be a function of month and hemisphere. And be applied only poleward of 60 north and south. If the blowing snow retrieval is outside of these latitudes, set this flag to zero regardless of month. Please see Table 4.5.

4.6.3 Blowing Snow Layer Confidence

When blowing snow has been detected, the blowing snow confidence flag will be constructed from a combination of the signal strength within the layer and the wind speed. We will compute the average scattering ratio (attenuated total backscatter divided by the attenuated molecular) within the layer multiplied by the surface wind speed:

$$\chi = \frac{\overline{\beta}}{\beta_m} U_s \quad (4.6.3)$$

Where U_s is the surface (10 m) wind speed. Table 4.4 shows the value of the blowing snow confidence flag as a function of χ . For version 8.3 of this document and ASAS code version 5.3, the value of χ will be written to the product (parameter *bsnow_intensity*) for both high rate and low rate.

Table 4.4. Blowing snow confidence flag

χ	Value	Confidence
< 20	1	None-little
>=20 - 50	2	Weak

>50 - 100	3	Moderate
>100 - 200	4	Moderate-high
>200 - 300	5	High
>300	6	Very high

**Table 4.5. Blowing Snow PSC interference flag
for latitudes poleward of 60 N/S**

Month	Southern Hemisphere	Northern Hemisphere
Jan	0	2
Feb	0	3
Mar	0	2
Apr	0	1
May	0	0
Jun	1	0
Jul	2	0
Aug	3	0
Sep	2	0
Oct	1	0
Nov	0	0
Dec	0	1

4.6.4 Blowing Snow Probability

Essery et al., 1999 present a method to predict the probability of the occurrence of blowing snow based on the 2m temperature (T , degrees Celsius), 10 meter wind speed (u_{10} m/s) and the age of the snow (A). We would like to include the results of this calculation in the ATL09 product. The blowing snow probability (ATL09 parameter $Bsnow_Prob$) is defined as:

$$P(u_{10}) = \left[1 + \exp \left\{ \frac{\sqrt{\pi}(\bar{u} - u_{10})}{\delta} \right\} \right]^{-1} \quad (4.6.4)$$

Where

$$\bar{u} = 11.2 + 0.365T + 0.00706T^2 + 0.9 \ln(A) \quad (4.6.5)$$

And

$$\delta = 4.3 + 0.145T + 0.00196T^2 \quad (4.6.6)$$

In Equation 4.6.5, A is the age of the snow on the ground in hours. Since this is very difficult to pin down, we will assume $A=6.0$. The blowing snow probability will be computed whenever the satellite is over a snow, sea ice or land ice surface. If the satellite is not over these surfaces, then the value should be set to invalid.

4.7 Apparent Surface Reflectance (ASR)

Apparent Surface Reflectance (ASR) is essentially a calculation of the received laser pulse energy from the surface divided by the transmitted laser energy multiplied by the two-way atmospheric transmission (T^2). In the case of a planetary body like the moon, which has no atmosphere, the ASR would be equal to the actual surface reflectance at the laser wavelength. On the earth however, the ASR is modified by the atmospheric transmission, which is in general not known. For a clear atmosphere, T^2 is about 0.81 at sea level (532 nm). Clouds and aerosols introduce further transmission loss ranging from a few tenths to a few orders of magnitude. This of course means that the ASR will always be less than the actual surface reflectance. For instance, if snow has a reflectance of 0.9 at 532 nm, then the ASR measured through a clear atmosphere at sea level will be 0.73 (0.81 x 0.9). If the surface reflectance is known well enough, the ratio of the apparent surface reflectance to the actual surface reflectance can be used as a relative measure of T^2 and thus as an indicator of the likely presence of clouds. This will be discussed further in section 4.6.1.

For ATLAS, ASR can be calculated as (following Yang et al., 2013):

$$\rho_{app} = \frac{\pi N_p r^2 D_c F}{NE A_t S_{ret}} \quad (4.7)$$

where N_p is the number of photons received from the surface, r is the distance between the satellite and the surface, D_c is the detector dead time correction factor, F is a calibration factor, analogous to but not the same as α in equation 2.11, E is the laser pulse energy, A_t is area of the telescope, and S_{ret} is the product of the transmittance of the optics and the quantum efficiency of the detector, and N is the number of laser pulse summed (400). Note that S_{ret} is known as the receiver return sensitivity and is the ATL03 parameter

/ancillary_data/atlas_engineering/receiver/rx_return_sensitivity and that each beam has its own sensitivity. This parameter contains the min, max, mean and standard deviation of the return sensitivity. We will use the mean value. The calibration factor F is required since we do not know the optical throughput of the system perfectly and it may change with time. It should be possible to calculate F by using data over the high Antarctic plateau where the surface reflectivity is fairly well known and the atmosphere is clear. In this region, ρ_{app} should have a value of approximately 0.95 times the two way transmission to the Antarctic plateau which can be calculated (approximately 0.88). We plan to do this using the first data acquired by ATLAS and then periodically re-evaluate its magnitude as the mission progresses. F will be an algorithm

adjustable parameter that is read in by the SIPS software (the final codes used to produce data products. SIPS is the Science Investigator Processing System). We will assume initially that $F = 1$. Note also that the above equation also assumes a lambertian surface reflectance. Update: after launch the value of F was determined to be between 0.50 and 0.60. For version 5.4 ASAS (version 4 data products), the value of F will be 0.56.

Note that using on orbit data, the approximate value of F above was determined to be 0.50. This was done by examining data over the Antarctic plateau where the maximum ASR is expected to be near 1.0, and data over the ocean in clear conditions where the ASR is expected to be near the ocean surface reflectance as determined from the 10 m wind speed (see Eq, 4.9).

Equation 4.7 was used to compute the ATLAS apparent surface reflectance as a function of received surface photons (N_p) (see Figure 4.6.1). For this computation, the following values were used: $r = 496$ km; $D_c = 1.1$; $F = 1.0$; $E = 160$ μ J; $A_t = 0.43$ m²; $S_{ret} = 3.79e17$, $N=400$. Note that A_{teles} assumes a telescope diameter 80 cm with a central obscuration of 30 cm.

In practice, the accuracy of the ASR calculation relies on the accuracy of surface detection; only if all photons from surface are detected, will we be able to determine the ASR accurately. In practice, this should be possible. The surface return will often come back in one atmospheric bin (30 m width), or at most 2-3 bins over very rough or highly sloping terrain.

We have two options for calculating N_p in Equation 4.7. The first is simply to use the *Surface Sig* parameter calculated and stored on ATL04 and read in by the ATL09 algorithm. The second way is to define N_p is using the ATL03 parameters *sig_count_high*, *sig_count_med* and *sig_count_low*. These parameters reside on the ATL04 product. We define a new control parameter called *Surface_Signal_Source* to select which surface signal to use for N_p in equation 4.8. *Surface_Signal_Source*=1 will mean to use the surface signal computed from the atmospheric histograms (parameter *Surface_Sig*). If *Surface_Signal_Source*=2, then use the sum of *sig_count_high* and *sig_count_med* to define N_p . NOTE that this sum will need to be dead time corrected (see section 2.1).

The transmitted laser energy for each beam, E should be known to within 5% and should be available at 25 Hz.

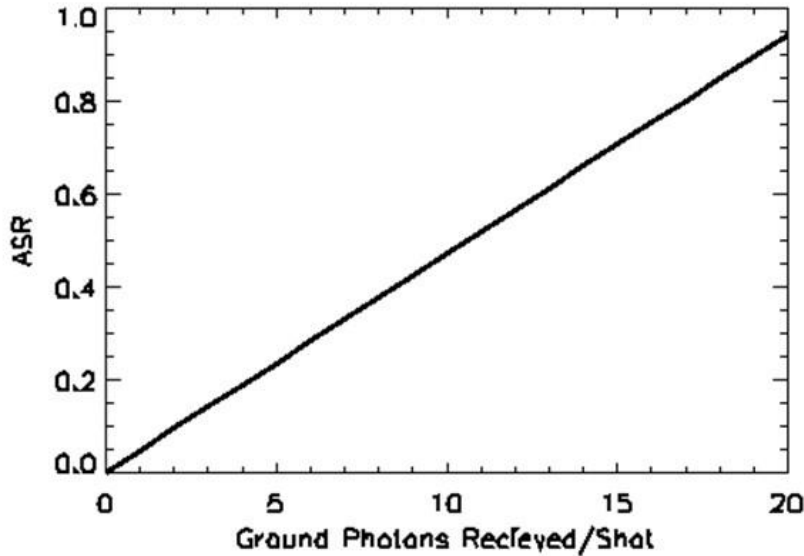


Figure 4.7.1 The apparent surface reflectance as computed from Equation 4.7 as a function of the number of surface return photons received per laser pulse. In practice, the ATLAS atmospheric data will be the sum of 400 laser pulses, and thus the photons received 400 times larger.

4.7.1 Cloud Detection using ASR

For a given surface and laser system, clouds lower the returning energy that is reflected by the surface (evidenced by a lower N_p) and hence lowers the ASR. Obviously, when clouds are present, ASR is a function of cloud optical depth (COD), but it is also a lesser function of cloud height and cloud microphysical properties. Figure 4.7.2 illustrates how cloud properties affect ASR. The black and gray curves are results of radiative transfer calculations with a 3D Monte Carlo model. Normalized by the clear-sky ASR, the results shown represent the two-way transmittance of the atmosphere. For example, a cloud with COD = 0.1 decreases the surface return by about 8% to 17%; while a cloud with COD = 1.0 decreases the surface return by 57% to 85%. The variability in surface return for a given COD is due to the variability in cloud altitude and microphysics. Fig. 4.7.2a shows that the lower the cloud, the higher the ASR. This is due to the fact that for lower clouds, photons that experienced multiple scattering have a larger probability of staying in the telescope field of view (Yang et al, 2010, 2011). Similarly, as shown in Figure 4.7.2b, everything else being equal, the larger the particle size, the higher is the ASR. This is because of larger particles' larger forward scattering probabilities; hence more photons tend to stay in the FOV.

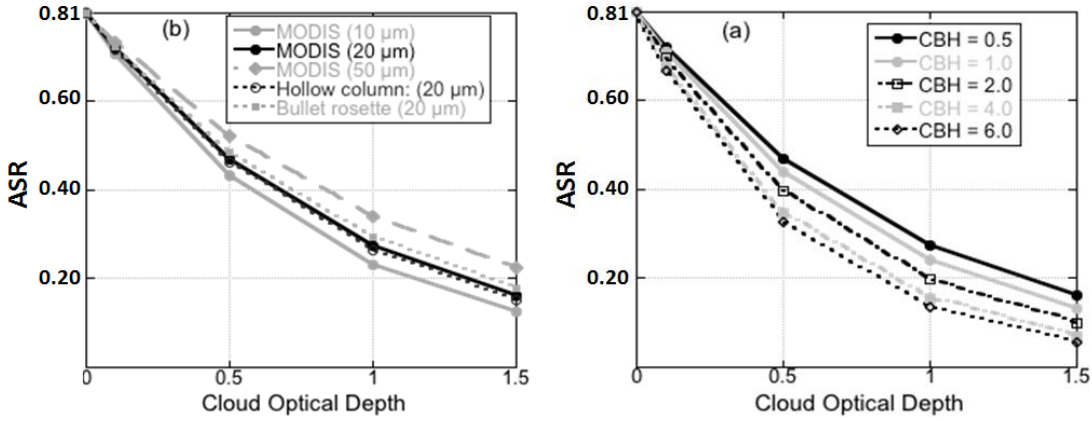


Figure 4.7.2. The effect of cloud properties on ICESat-2 ASR. Results are from Monte Carlo simulations over a surface of reflectivity 1. Telescope of field of view (FOV) is 40 m in diameter. (a) ASR changes as a function of COD for different cloud base heights (CBH) (in km). Cloud thickness is assumed to be 0.5 km; MODIS ice phase function for particle effective radius $r_e=20 \mu\text{m}$ is used. (b) Same as (a) but for different cloud particle sizes and shapes. The numbers in the *parentheses* in the legend are the effective radii. Clouds are assumed to be located at 0.5 - 1.0 km.

Overall, Figure 4.7.2 shows that clouds provide a strong signal in ASR that can be used for cloud detection. It is worth noticing that the cloud detection method presented here relies on the relative change of the ASR; hence absolute calibration of the instrument is not as important an issue, but the instrument stability can certainly affect the accuracy of the cloud detection results.

Since clouds can significantly reduce the ASR measured by the ICESat-2 detectors, it is possible to set a threshold to separate cloudy and clear conditions. However, if surface reflectivity varies significantly from location to location, cloud detectability will be lower compared to an otherwise uniform surface. To show this, we first examine how the ASR varies under clear sky conditions over the ice sheets. Figure 4.7.3a gives the results over Greenland, East and West Antarctica. Data are from the GLAS L2A campaign that began on 25 September and lasted until 19 November, 2003. All data for which clouds were detected have been removed.

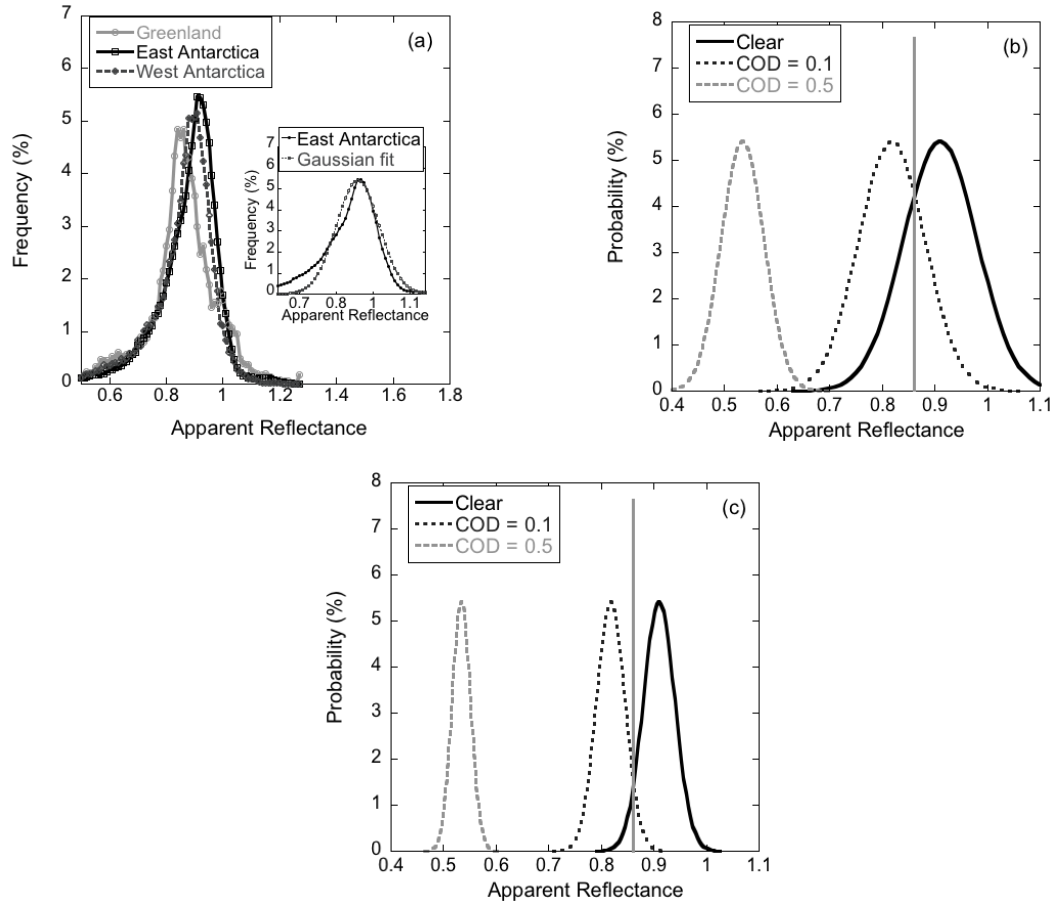


Figure 4.7.3 (a) Distribution of 1064 nm channel clear sky ASR. The inset is the Gaussian fit for the distribution over the East Antarctica. Data are from the GLAS L2A campaign. (b) Idealized ASR (Gaussian) for clear sky (mean = 0.91; standard deviation = 0.07) (solid black line), and the corresponding distributions for COD = 0.1 (dashed black line) and COD = 0.5 (dashed grey line). Also shown is an example threshold (solid gray line). (c) Same as (b), but for a Gaussian distribution with mean = 0.91 and standard deviation = 0.03.

The L2A campaign was selected because during this period GLAS had a fully functional atmosphere channel and the best cloud detection ability. Since the GLAS 532 nm channel is generally saturated by the surface signal, the ASR was derived from the 1064 nm channel measurements. It can be seen from the figure that the mode of the distribution from the Greenland ice sheet is lower than that of the Antarctica regions, with the highest from the East Antarctica.

The distributions shown in Figure 4.7.3a resemble a bell shape (Gaussian). To better illustrate the impact of surface reflectance variability on cloud detection, the clear sky distributions are fitted with Gaussian functions. The best fit for the distributions of Greenland, East and West Antarctica regions have means of 0.87, 0.91 and 0.89 and standard deviations of 0.07, 0.07 and 0.06, respectively. Figure 4.7.3b uses the Gaussian fit to the East Antarctica distribution (solid

black line) as an idealized ASR of snow cover under clear sky conditions. When clouds are present, the ASR will be smaller. For a plane parallel cloud, the shape of the distribution will stay Gaussian, only with a smaller mean and a smaller standard deviation. The change is a function of the cloud properties. For example, based on the radiative transfer simulations, for a cloud at 0.5 to 1.0 km with MODIS ice phase function for $r_e = 20 \mu\text{m}$, the ASR would decrease by 10% for COD = 0.1 and 41% for COD = 0.5. This example is plotted in Figure 4.7.3b with the dashed lines.

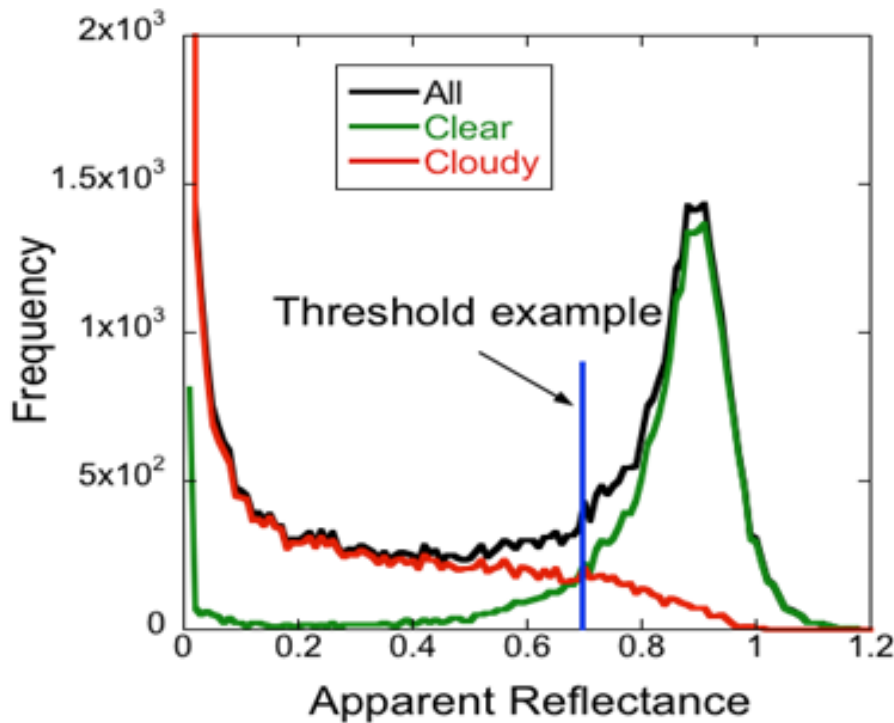


Figure 4.7.4. ASR distributions for clear (green), cloudy (red) and total (black) sky conditions over the West Antarctica ice sheet. Data are from the GLAS L2A campaign.

As can be seen in Figure 4.7.3b, there is an overlap between the ASR for clear sky and for cloudy sky (e.g. the curve for COD = 0.1). Hence it is no longer possible to completely separate clear and cloudy pixels with a single threshold: no matter where the threshold is set, some of the pixels will be misclassified. Generally, a smaller standard deviation in the ASR would indicate less overlap between the distributions under clear and cloudy sky conditions; hence allowing better cloud detection. Figure 4.7.3b and c demonstrate this point. The Gaussian distributions for clear sky in both panels have the same mean (0.91), yet their standard deviations are different, one is 0.07 (Figure 4.7.3b) and the other 0.03 (Figure 4.7.3c). As can be seen, the case with smaller standard deviation (Figure. 4.7.3c) corresponds to a smaller overlap, thus better cloud

detection. Obviously, for a given distribution of ASR, the misclassification rate is a function of the threshold and cloud optical depth and height.

Figure 4.7.4 shows an experiment of applying the ASR based cloud detection method to ICESat data. Again, the 1064 nm channel data from the GLAS L2A campaign is used. For this campaign, accurate cloud detection was achieved from the atmospheric channel (Palm et al 2002); hence the “truth” of which pixel is clear and which one is cloudy is known. The ASR distributions under clear and cloudy sky conditions are built from all the clear and cloudy pixels, respectively. As shown in the figure, an overlap exists between the distributions for clear and cloudy sky conditions; hence no threshold could separate all the clear pixels from the cloudy ones. The bimodal distribution shown in Figure 4.7.4 indicates that the ASR is a good test for cloud detection because misclassification can be minimized by putting the threshold in the valley region of the histogram. A sample threshold of 0.7 is marked on Figure 4.7.4. With this threshold, 15% of the clear pixels are misclassified as cloudy and 8% of the cloudy misclassified as clear. Certainly, the threshold can be adjusted to make the results either more cloud or clear conservative. As shown in Yang et al. (2013), the variability of the ASR at 532 nm is much smaller than that at the 1064 nm channel; hence cloud detectability will be strongly enhanced for the ATAS data.

4.7.2 ASR Cloud Detection Algorithm Implementation

The basic problem with the creation of an algorithm for cloud detection using ASR is the fact that for any given ATLAS observation the true surface reflectance (SR) is not well known. In order to detect the presence of cloud, we need to know the true surface reflectance (R_{true}) to within a certain tolerance. The accuracy with which we know the SR will determine the smallest cloud optical depth that can be detected. Based on model results, we feel the detection limit for this method will be optical depths in the range 0.2 to 0.4. For the first 4 data product releases (i.e. up through version 004) we used the best available data sets on the global surface reflectivity at 532 nm as a measure of R_{true} (for instance from GOME-2, Tilstra et al., 2017). For this latest data product version (005) we replace the GOME-2 data set with a global surface reflectance map created from actual ICESat-2 surface reflectance measurements acquired in only clear-sky conditions (i.e. when no cloud or aerosol layers are detected). ICESat-2 Apparent Surface Reflectance (ASR) measurements in clear-sky conditions are utilized from November 1, 2018 to October 31, 2020 to obtain a surface reflectance value on a 0.5 x 0.5 degree grid for each month of the year. This data set is then interpolated to a final global grid with 1/25 degree resolution. An example of the result for January and July is shown in Figure 4.7.5. As the mission progresses, and more data are obtained, we will update the monthly surface reflectance values utilizing the new data. This 1/25 degree global data set will be used to produce an ancillary file (ANC32) for use in the SIPS processing as described in the next section. We call this file the initial clear sky ASR climatology.

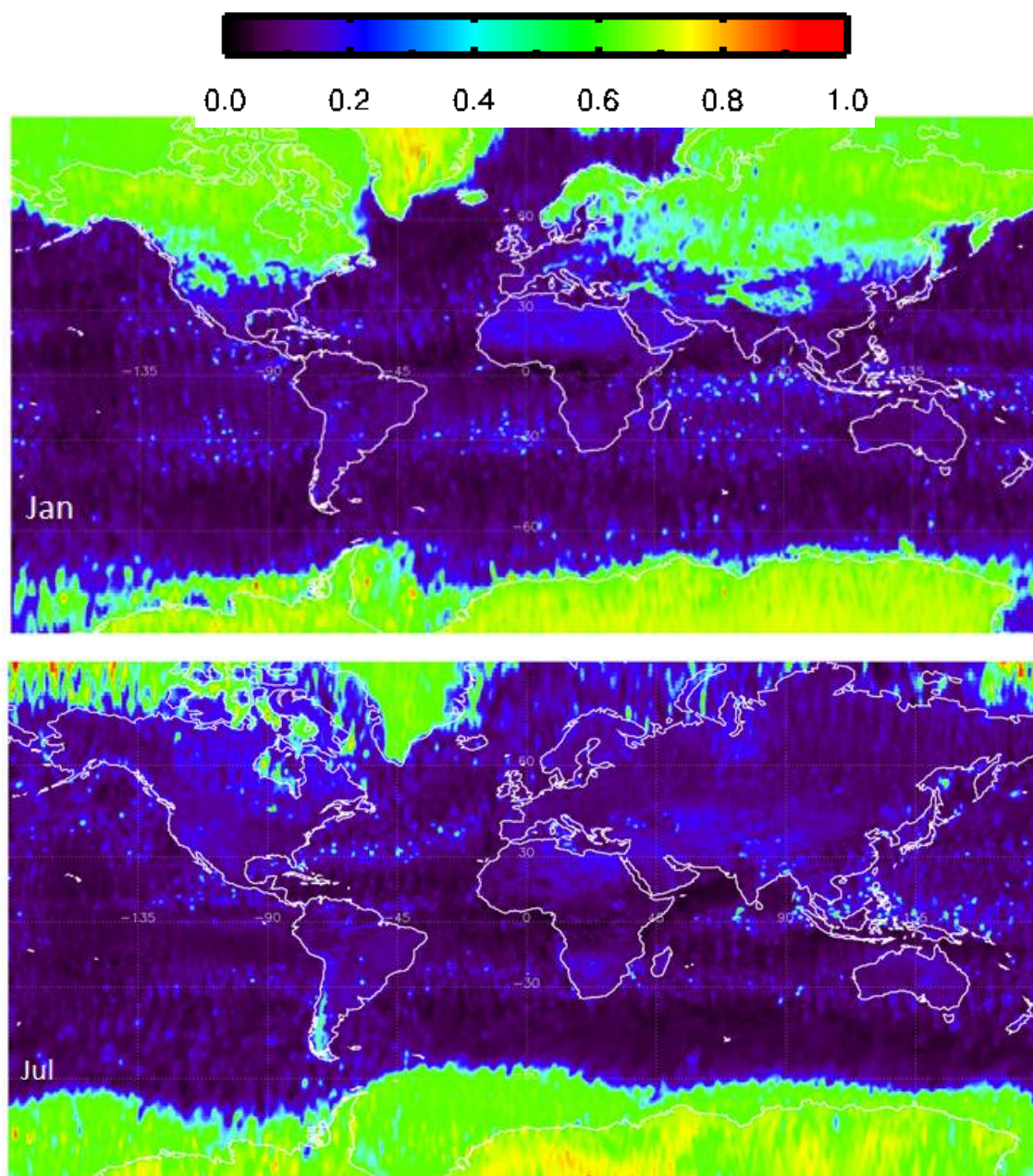


Figure 4.7.5. The monthly surface reflectivity climatology derived from ICESat-2 clear-sky ASR measurements for January and July.

4.7.2.1 ICESat-2 Clear Sky ASR Climatology

The clear sky ASR is a product of the true surface reflectance and two-way clear sky transmittance. Clear sky is defined as a molecular-only atmosphere (no clouds or aerosol detected). When GOME-2 data were used, they had to be multiplied by the two-way atmospheric

transmission. When using actual ICESat-2 measurements of clear-sky ASR, this is not required since it is inherent in the measurement as is ozone attenuation. This dataset is provided by the ATBD authors, and is produced following the procedure graphically shown in Figure 4.7.6:

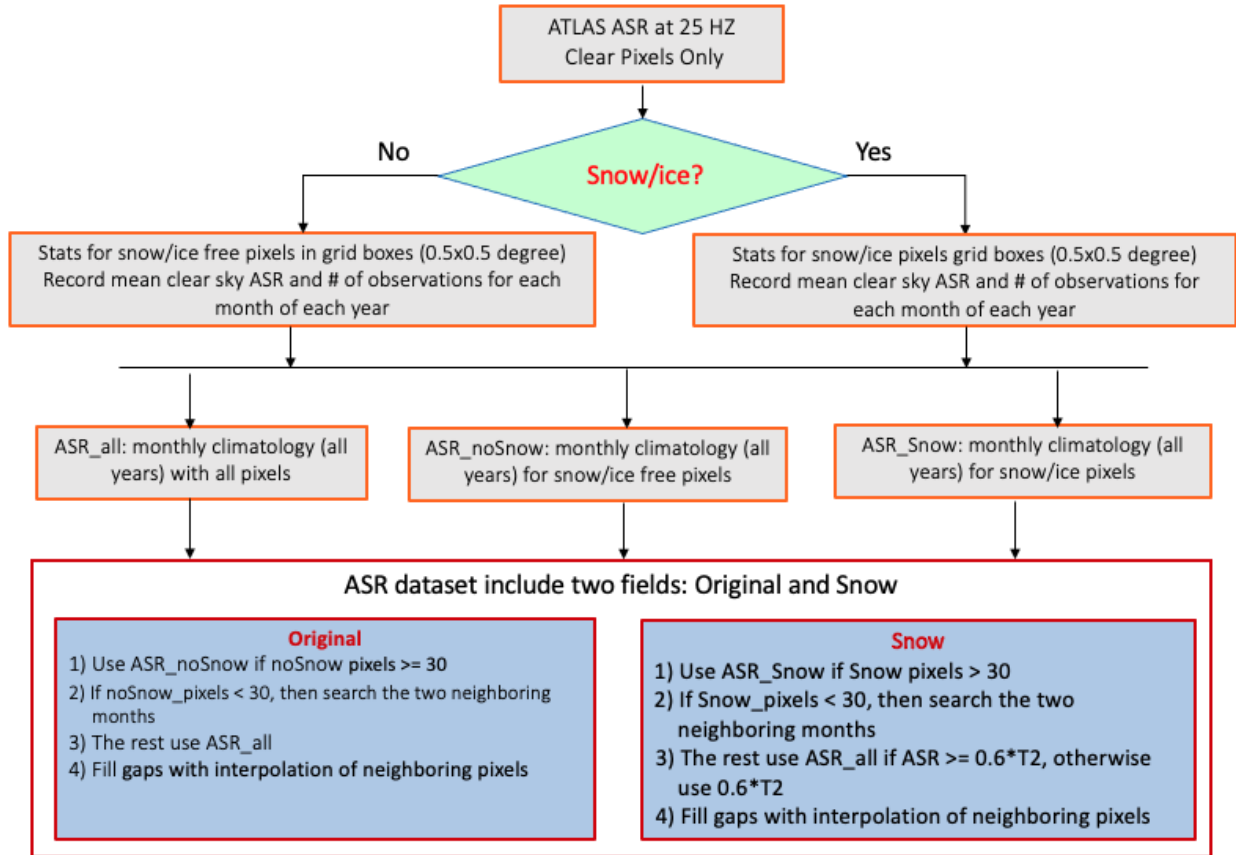


Figure 4.7.6. The procedure used to produce the clear-sky surface reflectivity using ICESat-2 ASR measurements. This is done on a 0.5 x 0.5 global grid and then interpolated to 1/25th degree grid for each month. Currently 2 years of ICESat-2 data are used.

After the above monthly global grids are obtained, they are interpolated to 1/25th degree resolution, The result will be a global data (over land/ice/snow only) set at 1/25 degree resolution with the 2 values defined - one for snow/ice and the other for snow-free land, which are stored in the dataset for each month of the year. This data set is called ACRtru. In the implementation of the SIPS code, these data sets will be used in conjunction with daily global snow cover data sets from NOAA (this is shown in Figure 4.7.7) and the surface type data set to obtain the value of clear sky ASR to use for cloud detection. The clear-sky ASR global (over land) grids will be re-generated each year as more data are collected.

The value of ACLR_{tru} is stored on the ATL09 product as parameter *Surf_Refl_True*.

Note: To obtain the latitude and longitude of each grid cell of the initial clear sky ASR climatology:

```

res = 1.0/25.0
nlat = long(180.0/res)
nlon = long(360.0/res)
lat = fltarr(nlat, nlon)
lon = fltarr(nlat, nlon)

for i = 0, nlat-1 do begin
  lat[i, *]=-90.0+(0.5+float(i))*res
endfor

for j = 0, nlon-1 do begin
  lon[*, j]=-180.0+(0.5+float(j))*res
endfor

```

http://www.star.nesdis.noaa.gov/smcd/emb/snow/HTML/multisensor_global_snow_ice.html

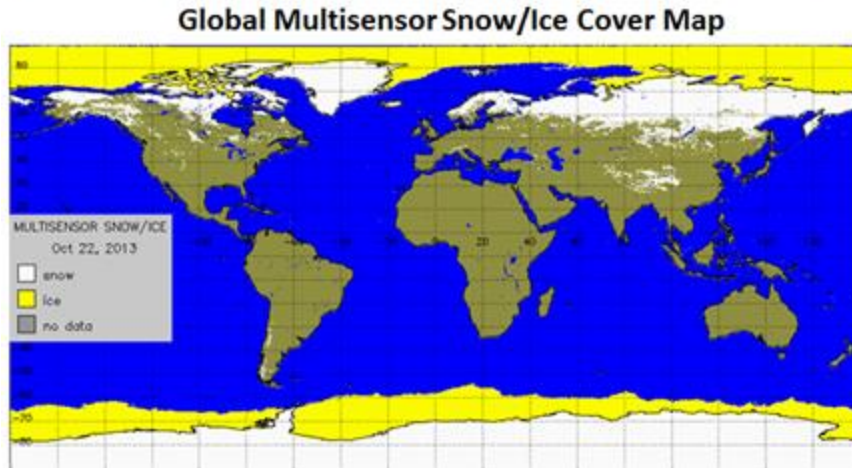


Figure 4.7.7. Example of the NOAA real-time snow and ice ancillary data set that will be used in selecting a value of the true surface reflectance (R_{true})

The snow and ice data can be obtained from:

http://satapsanone.nesdis.noaa.gov/northern_hemisphere_multisensor.html

also an IDL program can be found there to read the data.

4.7.2.2 Cloud Detection Using Apparent Surface Reflectance

Once we have the clear sky ASR, ACL_{true} , for the 4 seconds of observations, cloud detection is straightforward. Figure 4.7.8 gives the flow chart of the procedure. The details of each step are as follows:

- 1 Set ASR Cloud Threshold $T = ACL_{true} * \phi$, where ϕ is a factor for correcting the potential clear sky ASR biases. For ASAS 5.1 (version 001) products, ϕ was set to be 0.9

over land and 1.0 over ocean. For ASAS 5.2 (version 002) products we request that ϕ over land to be set to 1.10 and ϕ over ocean set to 0.90. For version 004, these have been changed to 1.1 over land and 1.0 over ocean.

- 2 For each of the 100 ASR values, Compute the probability (P) of cloud occurrence as defined in Figure 4.6.9. If $P \leq 40\%$, then label the point clear. If $P > 40\%$, label the point cloudy.
- 3 As shown in Table 4.5, assign a confidence flag to each of the cloud detection results.

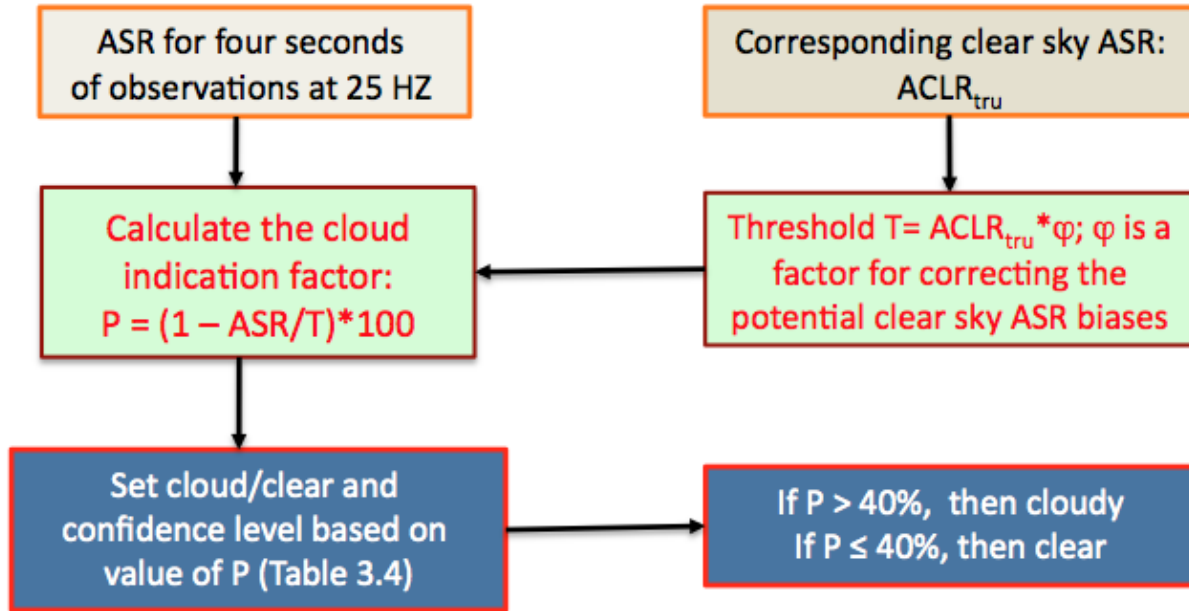


Figure 4.7.8. Flow chart for cloud determination for four seconds of data after the clear sky ASR is determined.

Using this procedure, if the measured ASR is less than $0.6 * T$ then the observation is defined as cloudy. The cloud probability (P) was not reported on the ATL09 product for ASAS 5.1 or 5.2 (versions 001, or 002), but rather was used to compute a 6 level flag as shown in Table 4.6. Figure 4.7.9 shows the viability of doing so with actual MABEL data. However, for ASAS 5.3 (version 003) the probability will be on the ATL09 product as parameter *asr_probability*.

Table 4.6. (Original Table 3.4) Cloud confidence flag (parameter *Cloud_Flag_ASR*).

$P = (1 - ASR/T) * 100$	Product Flag	Flag Meaning
80 – 100%	5	Cloudy with high confidence
60 – 80%	4	Cloudy with medium confidence
40 – 60%	3	Cloudy with low confidence
20 – 40%	2	Clear with low confidence
0 – 20%	1	Clear with medium confidence
$P < 0$	0	Clear with high confidence

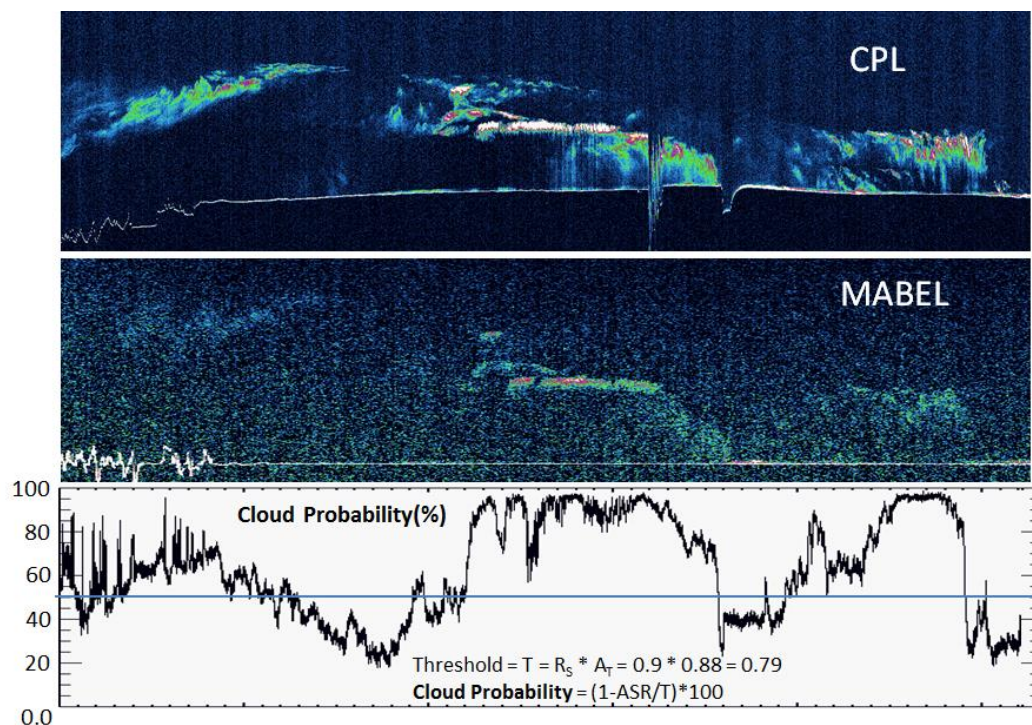


Figure 4.7.9 CPL (top) and MABEL (middle) for a scene over Greenland during the day. Example of cloud probability as computed from the MABEL measured apparent surface reflectivity and an assumed surface reflectivity of 0.90.

4.8 Ocean (or Open Water) Surface Reflectivity

Over water we can use the method of Lancaster et al. (2005) to compute the surface reflectance from surface wind speed (see Figure 4.8.1). Following Lancaster et al. (2005) the reflectance (R) of the ocean's surface is described by:

$$R = (1 - W)R_s + WR_f \quad (4.8.1)$$

Where R_s is the Fresnel reflectance from the surface, R_f is the reflection due to whitecaps and W is the fraction of the surface covered by whitecaps. Here we use $R_f = 0.22$ as the Lambertian reflectance of typical oceanic whitecaps at a wavelength of 532 nm (Koepke, 1984). Following Bufton et al. (1983), the Fresnel reflectance (R_s) is

$$R_s = \frac{\rho}{4\langle S^2 \rangle} \quad (4.8.2)$$

Where ρ is the Fresnel reflection coefficient and $\langle S^2 \rangle$ is the variance of the distribution of wave slopes. The Fresnel reflection coefficient is a function of wavelength and is computed as $\rho = 0.0205$ at 532 nm, from the tabulations of Hale and Querry [1973]. Cox and Munk (1954) provide an empirical description of $\langle S^2 \rangle$ as a function of wind speed:

$$\langle S^2 \rangle = 0.003 + 5.12 \times 10^{-3} U_{12.4} \quad (4.8.3)$$

Where $U_{12.4}$ is the wind speed at 12.4 m above the ocean surface. Numerical weather prediction models generally output wind speed at the 10 m height which can be adjusted to the 12.4 m level using equation 4.8.4 and assuming neutral atmospheric stability.

$$U_{12.4} = U_{10} \left(\frac{12.4}{10.0} \right)^{0.143} \quad (4.8.4)$$

Which assumes neutral stability. In computing the ocean lidar return from whitecaps the relative area of the ocean surface that they cover is estimated from the relation from Monahan and O'Muircheartaigh (1980):

$$W = 2.95 \times 10^{-6} U_{10}^{3.52} \quad (4.8.5)$$

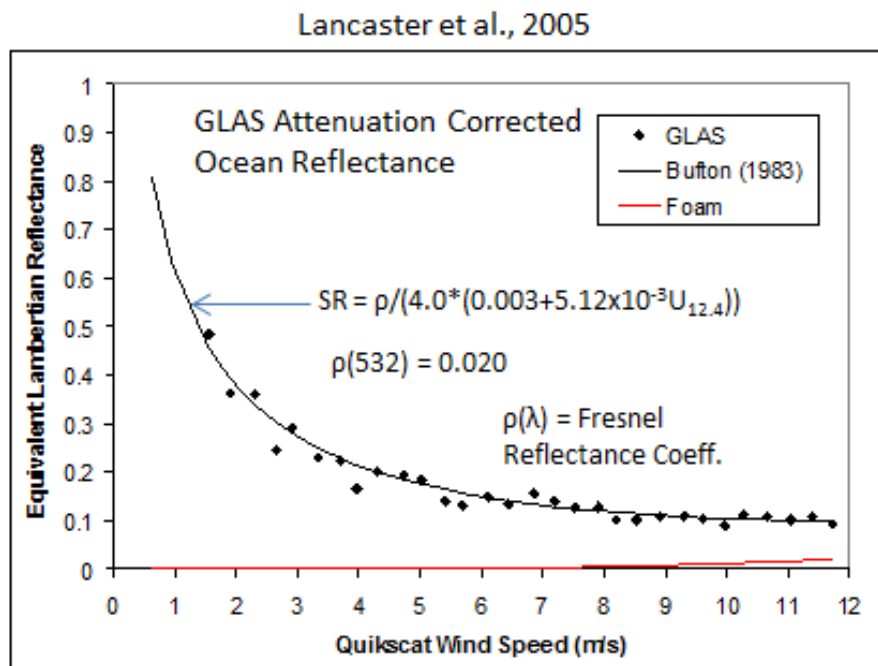


Figure 4.8.1. The theoretical ocean surface reflectivity curve (SR) for 1064 nm as a function of surface wind speed (solid black curve) and GLAS measured surface reflectivity (1064 nm) in areas known to be totally clear.

When the surface type flag indicates ocean or inland lakes, the above equations are used to compute the true surface reflectivity instead of the GOME-based climatological surface reflectivity data set.

Also (when over water), the computation of equation 4.8.1 shall be stored on the product as ATL09 parameter *Ocean_Surf_Reflec*.

4.9 Total Column Optical Depth Using ASR

The total atmosphere column particulate (not including molecular) optical depth can be computed from the apparent surface reflectance if the actual surface reflectance is well known. This condition holds over ocean where the surface reflectance can be computed from wind speed and over known surfaces like Antarctica and the interior of Greenland. Of course, for this method to be applicable, the surface return cannot be totally attenuated (zero). Thus, this technique is limited to cases where the overlying cloud and aerosol have a combined optical depth of less than about 3. Above that limit, the surface signal will be too small or totally attenuated (zero).

After computing R_{app} - the apparent surface reflectivity (at 25 Hz for all 3 beams) using equation 4.7, the true surface reflectance is computed as discussed in section 4.8 (if over water) or obtained from the GOME-based climatology of surface reflectance for the given surface type. Let the former be called R_{app} and the latter R_{true} . R_{app} must be corrected for molecular attenuation and the angle with which the laser beam makes with nadir (Θ):

$$R_{cor} = (R_{app}) / (\cos(\theta) \bar{T}_m^2) \quad (4.9.1)$$

where R_{cor} is the resultant corrected reflectance, θ is the tilt angle of the lidar with respect to nadir viewing (normally 0.1 but can reach 5.0 and may vary with laser beam), and \bar{T}_m^2 is the mean molecular two-way transmission for the entire atmospheric column at 532 nm (~0.81 at sea level). The correction for the tilt angle is usually very minor. The relationship between the corrected observed ATLAS reflectance (R_{cor}) and the true (or modeled) surface reflectance (R_{true}) is described below:

$$R_{cor} = R_{true} e^{-2\tau} \quad (4.9.2)$$

where τ is the optical depth of the particulates (cloud plus aerosol) in the atmospheric column. Solving for τ results in the equation:

$$\tau = -\frac{1}{2} \ln(R_{cor} / R_{true}) \quad (4.9.3)$$

which will be valid for all conditions where the ATLAS surface return is not saturated and where a surface signal is not totally extinguished by overlying clouds or aerosol. If the result of equation 4.9.3 is negative ($R_{cor} > R_{true}$), then set the resulting optical depth to 0.0. The result of equation 4.9.3 will be the ATL09 parameter *Column_OD_ASR*.

A quality flag for the total column optical depth (parameter name: *Column_OD_ASR_QF*) shall be produced which will be a function of the surface type. The highest quality value is given for those optical depths produced over ocean and ice sheets. The complete summary of this flag is shown in Table 4.7.

Table 4.7. Total Column Optical Depth from ASR Quality Flag (output parameter *Column_OD_ASR_QF*)

Surface Type	<i>Column_OD_ASR_QF</i>
No Surface Signal	0
Land	1
Sea Ice	2
Land Ice (Antarctica, Greenland)	3
Water	4

The Column_OD_ASR_QF will be computed using the NOAA flag and the IGBP flag.

5.0 Consolidated Cloud Flag

This section is added for Version 7.2. Here we aim to combine the various flags (*cloud_flag_atm*, *cloud_flag_ASR* and *bsnow_con*) into one flag. This is done to make it easier for the user to ascertain if clouds or blowing snow are present. The new ATL09 parameter will be called *layer_flag*. The overall logic for the flag values is shown in figure 5.1. For nighttime data, *cloud_flag_atm* will be of high quality. Thus if it is > 0 , (meaning at least one cloud layer was detected from the backscatter profiles), then *layer_flag* is set to 1. If no layers were found from the backscatter profiles, but blowing snow was found and its confidence level (*bsnow_con*) is greater than 2, then *layer_flag* is set to 1, otherwise *layer_flag* = 0.

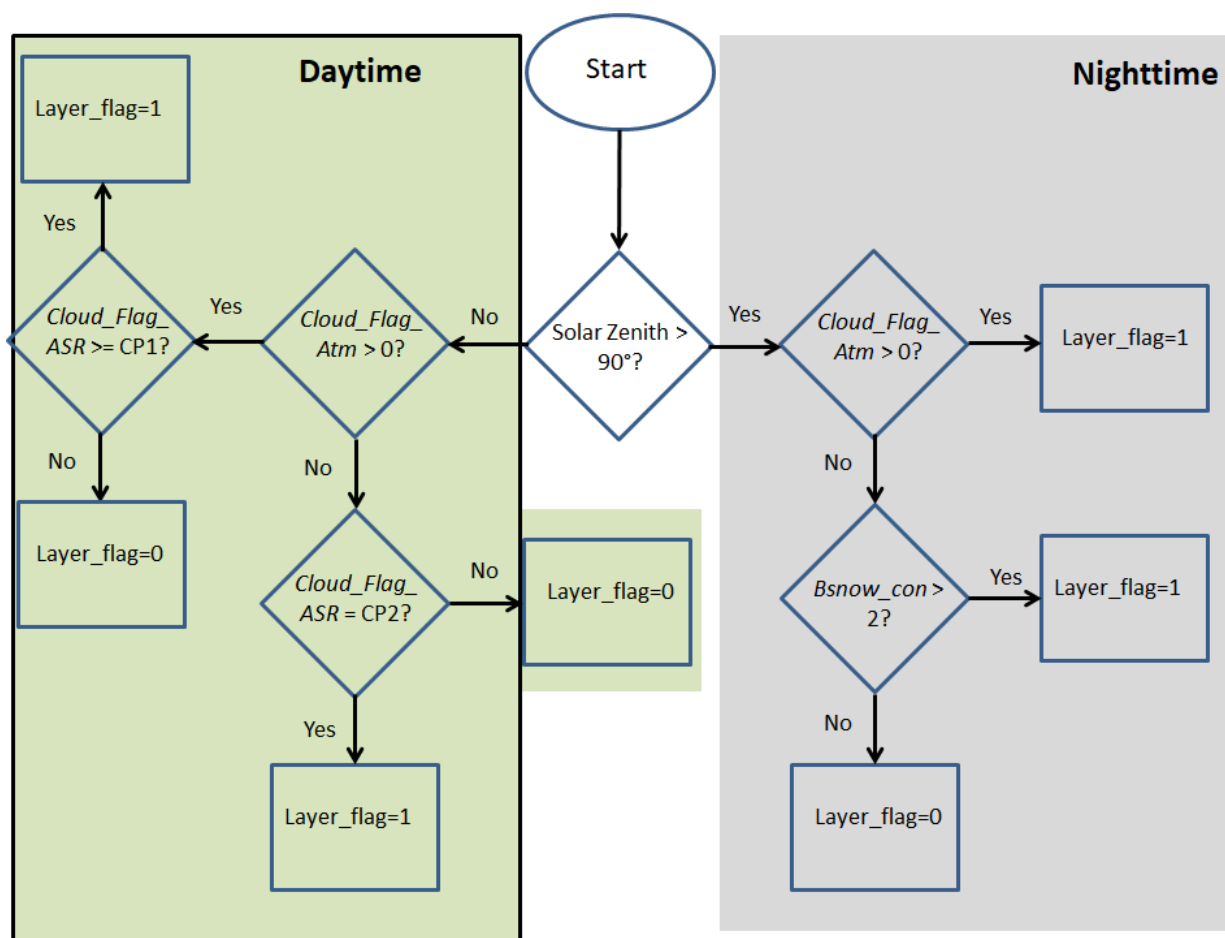


Figure 5.1 Logic flowchart for setting the ATL09 parameter *layer_flag*.

For daytime data, we again check *cloud_flag_atm*, and if it is > 0 we now check *cloud_flag_ASR* which is based on cloud detection from the apparent surface reflectivity. If its value is $\geq \text{CP1}$ (nominal value of 4), then we set *layer_flag* to 1, otherwise it is set to 0. If *cloud_flag_atm* = 0,

but *cloud_flag_ASR* = CP2 (nominal value of 5), then *layer_flag* = 1, otherwise it is 0. The factors CP1 and CP2 should be adjustable parameters read in from an ancillary file.

6.0 Quality Assessment

This section discusses the types of plots and images that can be made from each of the atmosphere products (ATL04 and ATL09) to help in the assessment of their quality. It is assumed that they will be created by the SCF team and viewed in the visualizer.

6.1 Calibration

To assess the accuracy and stability of calibration, a time series of the calculated calibration constants should be plotted daily and weekly. The plot should consist of the calculated constants from ATL04 and the time interpolated constants on ATL09 (produced from Equation 3.3) for each of the strong beams. Further, a plot of the average calibrated attenuated backscatter (as a function of height) for three latitude bands (poleward of 60 degrees north and south and from 60S to 60N) is to be plotted on the same graph as the average attenuated molecular backscatter for that latitude band. In the calculation of the average calibrated attenuated backscatter, profiles with clouds should be eliminated. These four plots should be done for each orbit. Note that the set of 4 plots should be made for each of the strong beams.

6.2 Background

The background is computed from the atmospheric profiles using two methods and reported on ATL04. Also on ATL04 is the 50 shot background computed onboard the instrument. Plots of background versus time or lat/lon should be generated for each orbit for all three methods plotted on the same plot. The solar zenith angle should also be plotted.

6.3 Surface Signal

A plot of the value of the surface signal should be made for each orbit. The surface signal (in photon counts) will be plotted as a function of latitude/longitude and show the percentage of time the surface was detected.

6.4 Calibrated Attenuated Backscatter

Images of scattering ratio should be generated. The scattering ratio is defined as the calibrated attenuated backscatter divided by the molecular backscatter. There should be two images for each orbit – one day (solar zenith angles less than 95 degrees) and one night (solar zenith angles greater than 95 degrees). The y axis will be height from -1 km to 20 km, and the x axis labeled with latitude and longitude. A map of the orbit track should also be provided. In the construction of the images, about 100 profiles (4 seconds) of data should be averaged to produce 1 image line.

6.5 Layer Heights

An image of calibrated attenuated backscatter should be made (of length ¼ orbit) and the layer heights superimposed on the image. Statistics on the percent cloudiness and the average number of layers found per orbit should be written on the plot.

6.6 Apparent Surface Reflectance

A plot of Apparent Surface Reflectance (ASR) as a function of lat/lon with a map of the orbit track shown on the same plot. Also images of calibrated backscatter with the ASR cloud probability flag (Table 3.4,

parameter *Cloud_Flag_ASR*) indicated on the image should be generated. Each image should span roughly ¼ of an orbit. This cloud flag could also be plotted on the layer height image (5.5 above) if feasible.

6.7 Ocean Surface Reflectance

A plot of the ocean surface reflectance (ATL09 parameter *Ocean_Surf_Reflec*) as a function of the 10 m wind speed should be made. Note that this parameter is only computed over water and will be invalid over land.

6.8 Total Column Optical Depth

The total column optical depth as computed from ASR (Equation 3.16, ATL09 parameter *Column_OD_ASR*) should be plotted for each orbit.

6.9 Multiple Scattering Warning Flag

The multiple scattering warning flag (parameter *msw_flag*) should be plotted for each orbit

7 Product Quality Parameters

The following are parameters that can be computed from the data that can aid in assessing the overall quality of the atmosphere products. Their magnitudes can be compared with a TBD range of values to assign a pass/fail type of grade to a given granule.

ATL04: 1) The number of calibration values and their average and standard deviation. 2) Average and standard deviation of the laser energy. 3) Average, min and max surface signal and the average of the retrieved surface height minus the DEM height. 4) Percent of time surface was detected. 5) Average, min and max background value computed from the 3 methods.

ATL09: 1) The average and standard deviation of the ratio of calibrated backscatter to molecular backscatter between 11 and 13.5 km (or top of profile) for all data poleward of 60 degrees north and south. 2) The min, max and average number of cloud layers detected from the DDA algorithm. 3) Percent clouds as detected by the DDA algorithm and from the Apparent Surface Reflectivity (ASR) method. 4) Min, max, average and standard deviation of ASR (parameter *apparent_surf_reflec*). Min, max and average of the ocean surface reflectance (parameter *ocean_surf_reflec*) and the total column optical depth (parameter *column_od_asr*). 5) Percent of time ground was detected.

For version 005, we introduce the composite POD/PPD flag that indicates the quality of input geolocation products for the specific ATL03 segment. While the pointing and orbit quality are not generally a concern to the atmosphere products, this flag will be useful for knowing when ocean scans are occurring as well as other calibration activities. This flag has been carried to ANC39 and should also be put on ATL04 and ATL09 (parameter *podppd_flag*). A non-zero value may indicate that geolocation solutions are degraded or that ATLAS is within a calibration scan period (CAL). Possible non-CAL values are: 0=NOMINAL; 1=POD_DEGRADE; 2=PPD_DEGRADE; 3=PODPPD_DEGRADE; possible CAL values are: 4=CAL_NOMINAL; 5=CAL_POD_DEGRADE; 6=CAL_PPD_DEGRADE; 7=CAL_PODPPD_DEGRADE

Appendix A.

pro pint2

```
; Program to interpolate pressure to lidar bin resolution from radiosonde data
; Note: Can be used with model data as well
; Inputs: p - pressure at radiosonde (model) levels (millibars)
;         t = temperature at radiosonde (model) levels (degrees Celsius).
;         z - height in meters of the p,t observation
; Output: pressure in millibars at the lidar bin resolution (delz, here = 30m)
; SPP 28Sep14
```

```
nlevs = 0
openr,1,'F:\Sondes\bermuda_sonde2' ; this is radiosonde data from Bermuda 09/22/14 12Z
readf,1,nlevs
p = fltarr(nlevs)
t = fltarr(nlevs)
z = fltarr(nlevs)
for i=0,nlevs-1 do begin
    readf,1,a,b,c
    p(i) = float(b)      ; pressure in mb
    z(i) = float(a) * 1000.0 ; height in m
    t(i) = float(c)      ; temperature in degrees Celsius
endfor
close,1
```

```
print,' nlevs = ', nlevs
t = t + 273.15          ; convert temperature to Kelvin
delz = 30.0d0          ; bin size (30 m)
top = z(nlevs-1)
bot = z(0)
print,'bot, top = ', bot, top
np = fix(top/delz) + 1
press = dblarr(np)
temp = dblarr(np)
height = fltarr(np)
ht = bot
layer = 1
bin = 1
press(0) = p(0)
```

```

layer_thickness = z(layer) - z(layer-1)
layer_bins = layer_thickness / delz
temp_delta = (t(layer) - t(layer-1)) / layer_bins
temp(0) = t(0)

x = double((z(1) - z(0)) / alog(p(0)/p(1))) ; this is the chi factor in ATBD eqn 2.4

while (ht lt top) do begin
  if (ht gt z(layer)) then begin
    layer++
    print,'layer = ',layer
    x = double((z(layer) - z(layer-1)) / alog(p(layer-1)/p(layer))) ; this is the chi factor in ATBD
eqn 2.4
    layer_thickness = z(layer) - z(layer-1)
    layer_bins = layer_thickness / delz
    temp_delta = (t(layer) - t(layer-1)) / layer_bins
  endif

  press(bin) = press(bin-1) * exp(-delz/x)
  temp(bin) = temp(bin-1) + temp_delta
  height(bin) = ht/1000.0
  ht = ht + delz
  bin++
endwhile

print, 'Number of', delz, 'm bins created: ',bin-1
;print,press

loadct,38,file='c:\Cpl\Config\colors.palm.tbl'

window,1
plot,press,height,background=255,color=0, xtitle='Pressure (mb)', ytitle='Height (km)'

oplot,temp,height, color=50
oplot,p,z/1000.0,psym=1,color=0,symsize=1
oplot,t,z/1000.0,psym=1,color=0,symsize=1

r = fltarr(256)
g = r

```

```

b = r
tvlct,r,g,b,/get
tvimage = tvrd()
help,tvimage
write_gif, 'F:\Sondes\pressure.jpg', tvimage, r,g,b

number_density = press * 1000.0 / (1.3806488e-16 * temp)
bmol = 5.450d0 * number_density * (550.0d0/532.0d0)^4 * 1.0d-26

window,3
plot,bmol,height,background=255,color=0, xtitle='532 nm Molecular Backscatter',ytitle='Height
(km)'

tvimage = tvrd()
help,tvimage
write_gif, 'F:\Sondes\molecular.jpg', tvimage, r,g,b

end

```

Appendix B.

Relation between geopotential height and geometric height

$$z = \frac{H}{G} \left(\frac{a}{a - \frac{H}{G}} \right)$$

Where:

z is the geometric height

H is the geopotential height

a = 6378137.0 m earth semi-major axis

$$G = \frac{\gamma(\phi)}{g_0}$$

$$g_0 = 9.81 \text{ ms}^{-2}$$

$$\gamma(\phi) = \gamma_e \frac{1 + k \sin^2 \phi}{\sqrt{1 - e^2 \sin^2 \phi}}$$

$$\gamma_e = 9.8321849378 \text{ ms}^{-2}$$

$$k = 0.00193185265241$$

$$e^2 = 6.69437999014 \times 10^{-3}$$

$$\phi = \text{latitude}$$

Reference: Lambert et al., 1999

And

<http://www.ofcm.gov/fmh3/pdf/12-app-d.pdf>

Appendix C

pro backm1_v2, profile, solar_elevation, latitude, longitude, background

; SPP December, 2018

; This routine computes the background from the raw photon count atmospheric data

; Inputs: profile - 2 dimensional array (x,y), where x is the number of bins (467) and y is the number of profiles

; solar_elevation - Solar elevation angle. One dimensional array which usually contains less elements than (y) above.

; Output: Background for each of the input y profiles.

; Steve Palm, December 2018

; *** these next 14 parameters should be ASAS adjustable constants**

backg_night_scale_factor = 1.00600 ; decreasing this factor adds to signal, or makes avg profile more vertical

backg_day_scale_factor1 = 0.0100

```

backg_day_scale_factor2 = 0.990
backg_min_solar_elev = -7.0
backg_max_solar_elev = -1.0
backg_night_background = 0.0600 ; making this larger makes avg profile more horizontal, smaller,
more vertical
backg_twilight_scale_factor = 0.6
backg_nseg = 6 ; backg_nseg is the atl04 parameter used in the current background method 1 and 2.
backg_day_exp_factor = 8.5
backg_response_time = 10.0e-9

saa_latmin = -40.0
saa_latmax = -0.0
saa_lonmin = -73.0
saa_lonmax = -20.0

,*****

rpd = 3.14159265 / 180.00
nshot_sum = 400.0 ; number of ATLAS laser shots summed in profile

r = size(profile)
nbt = r(1)
nb = r(1) - 33 ; set number of bins to 1 km above the bottom of profile
nrecs = r(2) ; number of atm profiles

solar_elev = congrid(solar_elevation,nrecs,/interp) ; interpolate 1 Hz solar elevation to 25 Hz
;r = size(solar_elevation)
;nsa = r(1) ; number of solar elevation angles (these are at 1 Hz, while profiles are 25 Hz).
;pfac = float(nrecs) / float(nsa)

seg_len = nb / backg_nseg

for rec = 0,nrecs-1 do begin

    ptr = rec
    ; ptr = fix(float(rec)/pfac) ; ptr is used to index into the solar elevation array to get an angle for the
current 25 Hz atm profile
    ; if (ptr ge nsa) then ptr = nsa-1

    if (solar_elev(ptr) gt backg_max_solar_elev) then begin
        sum = fltarr(backg_nseg)

```

```

for i=0,backg_nseg-1 do begin
  b1 = i * seg_len
  b2 = b1 + seg_len
  sum(i) = 0.0
  for bin=b1,b2 do begin
    sum(i) = sum(i) + float(profile(bin,rec))
  endfor
endfor

sval = min(sum,idx) / float(seg_len+1)
dtfac = 1.0 / (1.000 - backg_response_time * sval / 0.20e-6 / nshot_sum)
fac = dtfac^backg_day_exp_factor
background(rec) = sval ; set daytime background as average of minimum segment
if (background(rec) lt backg_night_background) then background(rec) = backg_night_background
badd = background(rec) * backg_day_scale_factor1
background(rec) = background(rec) + (badd / fac)
endif else begin

if (solar_elev(ptr) le backg_min_solar_elev) then begin ; nighttime
  background(rec) = backg_night_background * backg_night_scale_factor
  if (latitude(ptr) gt saa_latmin and latitude(ptr) lt saa_latmax) then $ ; this is for the south atlantic
  anomaly
  if (longitude(ptr) gt saa_lonmin and longitude(ptr) lt saa_lonmax) then begin
    b1 = total(profile(0:16,rec), 1) / 17.0 * 0.6
    b2 = total(profile(nbt-17:nbt-1,rec), 1) / 17.0 * 0.6
    background(rec) = b1
    if (b2 lt b1) then background(rec) = b2
  endif
endif else begin ;solar elevation <= backg_max_solar_elev and > backg_min_solar_elev:
twilight
  fac = cos(8.0*solar_elev(ptr)*rpd) * 1.05
  b1 = total(profile(0:16,rec), 1) / 17.0 * fac
  b2 = total(profile(nbt-17:nbt-1,rec), 1) / 17.0 * fac
  sval = b1
  if (b2 lt b1) then sval = b2
  dtfac = 1.0 / (1.000 - backg_response_time * sval / 0.20e-6 / nshot_sum)
  fac = dtfac^backg_day_exp_factor
  background(rec) = sval
  if (background(rec) lt backg_night_background) then background(rec) =
backg_night_background
  badd = background(rec) * backg_day_scale_factor1
  background(rec) = background(rec) + (badd / fac)

```



```

;    background(rec) = total(profile(0:32,rec), 1) / 33.0 * backg_twilight_scale_factor
    if (background(rec) < backg_night_background) then background(rec) =
backg_night_background
    endelse

endelse

endfor

end

```

Appendix D

```

pro atm_find_ground_v4, raw_profile, solar_elev, dem_h, profile_top_ht, fgrnd, gheight, gbin,
surface_signal, backg_subtracted_profile, thr1, thr2, $
surface_bin, i1, i2, wide_ground

```

; Program to find the ground return in ATLAS raw atmospheric histograms (30 m bins, 467 bins total).
; SPP, April 2020

```

; Inputs: raw_profile - the 467 bin raw atmospheric histogram (photons)
;    dem_h - Digital elevation model surface height associated with the location of this atm histogram
(m)
;    profile_top_ht - The height above the ellipsoid of the top of the atmospheric histogram (m)

; Output: fgrnd - a -1/0/1 flag indicating if the surface was found (1) or not (0), or could not search (-1)
;    gheight - the height above the ellipsoid of the surface return if found. If not found, or if unable to
;    search for it, this equals -1000.0 (m)

```

```

Nwin = fix(10.0/0.030)
r = size(raw_profile)
nbins = r(1)

```

```

; get estimate of the background level
range_to_ground = profile_top_ht - dem_h
ground_bin = fix(range_to_ground / 30.0)

```

```

lev = 10
sdfac = 6.0
backg = 0.0

```

```

stdev = 1.0
if (ground_bin lt nbins-1) then begin
  nseg = 5
  nbseg = ground_bin / nseg
  min_seg_avg = 10000.0
  for i=0,nseg-1 do begin
    i1 = i * nbseg
    i2 = i1 + nbseg
    r = moment(raw_profile(i1:i2))
    seg_avg = r(0)
    seg_std = sqrt(r(1))
    if (seg_avg lt min_seg_avg) then begin
      min_seg_avg = seg_avg
      min_seg_std = seg_std
    endif
  endfor

  backg = min_seg_avg
  stdev = min_seg_std
  lev = min_seg_avg + min_seg_std * 3.0
  ptr = where(raw_profile(0:ground_bin-10) gt lev)
  raw_profile(ptr) = backg

  if (solar_elev lt 0.0) then begin
    sdfac = 6.0
;   print,'solar_elev, backg = ', solar_elev, backg
  endif else begin
    sdfac = 3.0
  endelse
endif else begin
  print,'Returning from find ground routine. ground_bin, nbins = ', ground_bin, nbins
  fgrnd = -1
  return
endelse

backg_subtracted_profile = raw_profile - backg

; set thresholds for surface signal
thr1 = sdfac * stdev
if (thr1 lt 3.0) then thr1 = 3.0
thr2 = sdfac * stdev * 20.0
if (thr2 gt 1000.0) then thr2 = 1000.0

```

```
; use DEM and profile top ht to define a search window
range_to_ground = profile_top_ht - dem_h
ground_bin = fix(range_to_ground / 30.0)
;print,'range_to_ground,ground_bin,dem_h = ', range_to_ground,ground_bin,dem_h,profile_top_ht
;stop
```

```
fgrnd = 0
gheight = -1000.0
gbin = -1
surface_signal = 0.0
surface_bin = 0
wide_ground = 0
```

```
if (ground_bin gt nbins-Nwin and ground_bin lt nbins-1) then begin
```

```
    i1 = ground_bin - 5      ; define a search window around the dem value of the surface height
    i2 = ground_bin + 4
    if (i2 ge nbins) then i2 = nbins - 1
```

```
; print,'i1, i2 = ', i1, i2
```

```
    i = i2
    while (i ge i1) do begin
        if (backg_subtracted_profile(i) gt thr1) then begin
            max_sig0 = backg_subtracted_profile(i)
        ;   max_sig0 = raw_profile(i)
            max_bin0 = i
            surface_bin = max_bin0
            fgrnd = 1
            i = 0
        endif
        i--
    endwhile
```

```
;print,'mbin = ', mbin
```

```
if (fgrnd gt 0) then begin
    max_sig1 = 0.0
    max_sig2 = 0.0
    max_sig3 = 0.0
```

```

mx = max(backg_subtracted_profile(max_bin0-5:max_bin0),im)
max_bin = max_bin0-5 + im
if (max_bin ne max_bin0) then begin
    max_sig1 = raw_profile(max_bin)
    max_sig1 = backg_subtracted_profile(max_bin)
    surface_bin = max_bin
endif
thr2 = mx * 0.20
if (thr2 lt 20.0*thr1) then thr2 = 20.0*thr1
if (backg_subtracted_profile(max_bin-1) gt thr2) then begin
    max_sig2 = raw_profile(max_bin-1)
    max_sig2 = backg_subtracted_profile(max_bin-1)
    surface_bin = max_bin - 1
    wide_ground = 1
endif

if (backg_subtracted_profile(max_bin-2) gt thr2) then begin
    wide_ground++
    max_sig3 = raw_profile(max_bin-2)
    surface_bin = max_bin - 2
endif

; if (wide_ground ge 1) then $
;   print,'thr2 = ', thr1,thr2,mx,solar_elev

gheight = profile_top_ht - surface_bin*30.0
surface_signal = max_sig0 + max_sig1 + max_sig2 + max_sig3
gbin = surface_bin

endif

endif else begin          ; calculated ground bin is out of range - could not perform search
    fgrnd = -1
endelse

```

References

Bodhaine B. A., N. B. Wood, E. G. Dutton, and J. R. Slusser: "On Rayleigh optical depth calculations", J. Atmos. Ocean Technol., 16, 1854-1861 (1999) and/or the CALIPSO level 1 ATBD

- Bufton, J.L., F.E. Hoge and R.N. Swift, (1983), Airborne measurements of laser backscatter from the ocean surface. *Appl. Opt.*, **22**, **17**, <https://doi.org/10.1364/AO.22.002603>
- Cox, C. and W. Munk, (1954), Measurement of the Roughness of the Sea Surface from Photographs of the Sun's Glitter. *J. Opt. Soc. Am.* **44**, 838-850
- Essery, R., L. Long and J. Pomeroy, 1999, A distributed model of blowing snow over complex terrain. *Hydrol. Process.* **13**, 2423±2438 (1999)
- Hale, G.M. and M.R. Querry, (1973), Optical constants of water in the 200-nm to 200-µm wavelength region. *Appl. Opt.*, **12**, **3**, <https://doi.org/10.1364/AO.12.000555>
- Iqbal, M., An Introduction to Solar Radiation, Academic Press, New York, NY, 1983.
- Ismail, S. and E. Browell, 1989: Airborne and spaceborne lidar measurements of water vapor profiles: a sensitivity analysis. *Appl. Opt.*, **28**, 3603-3615
- Koelemeijer, R.B.A., J.F. de Haan and P. Stammes, (2003), A database of spectral surface reflectivity in the range 335–772 nm derived from 5.5 years of GOME observations. *J. Geophys. Res. Atmos.* <https://doi.org/10.1029/2002JD002429>
- Koepke, P., (1984), Effective reflectance of oceanic whitecaps. *Appl. Opt.*, **23**, **11**, <https://doi.org/10.1364/AO.23.001816>
- Lambert, A., P.L. Bailey, D.P. Edwards, J.C. Gille, C.M. Halvorson, B.R. Johnson, S.T. Massie and K.A. Stone, 1999, High Resolution Dynamics Limb Sounder Level-2 Algorithm Theoretical Basis Document. http://disc.sci.gsfc.nasa.gov/repository/Mission/HIRDLS/3.3_Product_Documentation/3.3.4_Product_Gen_Algorithms/sw-hir-339.pdf <http://www.ofcm.gov/fmh3/pdf/12-app-d.pdf>
- Lancaster, R.S., J.D. Spinhirne and S.P. Palm, (2005), Laser pulse reflectance of the ocean surface from the GLAS satellite lidar. *Geophys. Res. Lett.* **32**, **22**, <https://doi.org/10.1029/2005GL023732>
- Martino, A., J., T. A. Neumann, N. T. Kurtz, and D. McClellan, (2019), ICESat-2 Mission Overview and Early Performance. Proc. SPIE 11151, Sensors, Systems, and Next-Generation Satellites XXIII, 111510C (10 October 2019); <https://doi.org/10.1117/12.2534938>
- McGill, M., D. Hlavka, W. Hart, V.S. Scott, J. Spinhirne and B. Schmid, (2002), Cloud physics lidar: Instrument description and initial measurement results. *Appl. Opt.*, **41**, **18**, 3725-3734, <https://doi.org/10.1364/AO.41.003725>
- Monahan, E.C. and I. Muircheartaigh, (1980), Optimal power-law description of oceanic whitecap dependence on wind speed. *J. Phys. Oceanogr.*, **10**, 2094-2099. [https://doi.org/10.1175/1520-0485\(1980\)010%3C2094:OPLDOO%3E2.0.CO;2](https://doi.org/10.1175/1520-0485(1980)010%3C2094:OPLDOO%3E2.0.CO;2)
- Palm, S. P., Y. Yang and U. Herzfeld, 2020: Ice, Cloud, and Land Elevation Satellite (ICESat-2) Project Algorithm Theoretical Basis Document for the Atmosphere, Part I: Level 2 and 3 Data Products, version 3 DOI 10.5067/SOAMP4TOYLWJ

She, C., 2001: Spectral structure of laser light scattering revisited: Bandwidths of nonresonant scattering lidars. *Appl. Opt.*, 40, 4875–4884

Spinhirne, J.D., S.P. Palm, W.D. Hart, D.L. Hlavka and E.J. Welton, (2005) Cloud and aerosol measurements from GAS: Overview and initial results. *Geophys. Res. Lett.*, **32**, **22**,
<https://doi.org/10.1029/2005GL023507>

Vigroux, E., Contribution a l'etude experimentale de l'absorption de l'ozone, *Ann. Phys.*, 8, 709-761, 1953.